CELLULAR AUTOMATA - SIMPLICITY BEHIND COMPLEXITY

Edited by **Alejandro Salcido**

INTECHWEB.ORG

Cellular Automata - Simplicity Behind Complexity
Edited by Alejandro Salcido

Published by InTech
Janeza Trdine 9, 51000 Rijeka, Croatia

Publishing Process Manager Iva Lipovic
Technical Editor Teodora Smiljanic
Cover Designer Martina Sirotic
Image Copyright Shebeko, 2010. Used under license from Shutterstock.com

First published March, 2011
Printed in India

A free online edition of this book is available at www.intechopen.com
Additional hard copies can be obtained from orders@intechweb.org

Cellular Automata - Simplicity Behind Complexity, Edited by Alejandro Salcido
p. cm.
ISBN 978-953-307-230-2

Contents

Preface

In the early 1950s, at the suggestion of Stanislaw Ulam, John Von Neumann introduced the cellular automata as simple mathematical models to investigate self-organisation and self-reproduction. Cellular automata make up a very important class of completely discrete dynamical systems. The physical environment of cellular automata is constituted of a finite-dimensional lattice, with each site having a finite number of discrete states. The evolution in time of a cellular automaton goes on in discrete steps, and its dynamics is specified by some local transition rule, fixed and definite. In spite of their conceptual simplicity, which allows for an easiness of implementation for computer simulation, and a detailed and complete mathematical analysis in principle, the cellular automata systems are able to exhibit a wide variety of amazingly complex behavior. This feature of *simplicity behind complexity* of cellular automata has attracted the researchers' attention from a wide range of divergent fields of study of science, which extends from the exact disciplines of mathematical physics up to the social ones, and beyond. In fact, nowadays, cellular automata are a core subject in the sciences of complexity. Thus, numerous complex systems containing many discrete elements with local interactions, and their complex collective behaviour which emerge from the interaction of a multitude of simple individuals, have been and are being conveniently modelled as cellular automata. For example, the dynamical Ising model, gas and fluid dynamics, traffic flow, various biological issues, growth of crystals, nonlinear chemical systems, land use and population phenomena and many others. Moreover, cellular automata are not the only models in natural sciences such as biology, chemistry and physics, but they are also, thanks to their complete space-time and state discreteness, appropriate models of parallel computation. Thus, cellular automata permit descriptions of natural processes in computational terms (computational biology, computational physics), but also of computation in biological and physical terms (artificial life, physics of computation).

In this book the versatility of cellular automata for modelling a wide diversity of complex systems is underlined through the study of a number of outstanding problems with the cellular automata innovative techniques. This book comprises twenty five contributions organized in four main sections: *Land Use and Populations Dynamics; Dynamics of Traffic and Network Systems; Dynamics of Social and Economic Systems; and Statistical Physics and Complexity*. Brief descriptions of the book chapters are presented in the following paragraphs.

Land Use and Populations Dynamics. Chapter 1 describes a semi-automated, interactive method that was designed and implemented to dynamically create transition

rules and calibrate a land-use CA model. The proposed method combines the benefits of conditional and mathematical rules and is adaptable in terms of number of land-use classes, and spatial and temporal scale of the input data. Chapter 2 presents and describes a cellular automata model for simulating the population distribution of the city of Vienna from 1888 to 2001. It has also developed a sensible and robust concept for the explanation of the driving forces of urban development processes, and it was shown that the development of the population density can be essentially regulated by infrastructure investments. In Chapter 3, the deforestation processes in a region called São Félix do Xingu, located in east-central Amazon, are simulated with a cellular automata model called Dinamica EGO. It consists of an environment that embodies neighbourhood-based transition algorithms and spatial feedback approaches in a stochastic multi-step simulation framework. The modelling experiment demonstrated the suitability of the adopted model to simulate processes of forest conversion, unravelling the relationships between site attributes and deforestation in the area under analysis. Chapter 4 demonstrates that the heuristic search methods for the solution of spatial optimization problems have to be designed in accordance with the spatial character of the field under study, which can be fittingly modelled by means of cellular automata. Two basic approaches are presented in this chapter to pursue a balance between local and global characteristics. Chapter 5 demonstrates the potential of cellular automata as a tool for urban planning and development using two models and case studies, one from Saudi Arabia and the other from the Republic of Korea. The strengths and weaknesses of the models are discussed, including areas for further development. Chapter 6 presents three cellular automata that simulate the behavior of the population dynamics of three biological systems. The first one deals with artificially-living fish divided into two groups: sharks (predators) and fish that are part of their food chain (preys). The second model introduces a simulation of the HIV evolution in the blood stream of positive individuals with no antiretroviral therapy. The last model extends the previous one and considers the HIV dynamics in individuals subject to medical treatment and the monitoring of the medication potency and treatment adhesion. Finally, Chapter 7 explores some of the fundaments of cellular automata models and the reasons why these are being so widely applied nowadays, particularly to urban systems and ecology, all of which seem to be connected directly to the fact that the transport equations are common as much to the socioeconomic phenomena as to physics.

Dynamics of Traffic and Network Systems. Chapter 8 presents an overview of the basic cellular automata models for traffic flow. A maximum entropy approach for analyzing the equilibrium properties of the cellular automata models for multi-speed traffic flow in a single lane highway is also proposed and discussed. It is shown, in particular, that the traffic cellular automata models of Nagel-Schreckenberg and Fukui-Ishibashi evolve rapidly towards steady states very close to equilibrium. In Chapter 9, a modified model of the car-following Nagel-Schreckenberg model is proposed by incorporating the agent and diligent driver into it. The modified evaluation of the proposed parameter, the fundamental diagram, spatio-temporal patterns, effect of lane-changing and car-following with respect to the evacuation time, combination parameter of diligent and agent driver in the case of evacuation time and the effectiveness are investigated. Chapter 10 presents a simple cellular automaton model to study the typical bus dynamics in a modern city. At a first stage, the nontrivial fluctuations are prescribed by the stochastic moving of bus interacted with the stochastic arrival of passengers, and at a second stage, the bus schedule interrupted by the traffic lights is examined. The city

buses time headway distribution is analyzed and compared against real time headway measurements. Chapter 11 studies the traffic flow dynamics with real-time information. The influence of feedback strategies is introduced, based on a two-route scenario in which dynamic information can be generated and displayed on the board to guide road users to make a choice. The model incorporates the effects of adaptability into the traffic cellular automata. Simulations demonstrate that adopting these optimal information feedback strategies provide a high efficiency in controlling spatial distribution of traffic patterns. Chapter 12 presents the application of the cellular automata paradigm for modelling network systems. The combination of cellular automata and graph structure was successfully applied for simulating phenomena that belong to general class of network systems located in consuming or producing environment. Two examples were investigated, anastomosing river systems and vascular systems created in processes of tumor induced angiogenesis, showing how broad meaning cellular automata now has. Chapter 13 shows that cellular automata modelling technique could partially fill the gap in describing the dynamics of biomolecular networks. While not able to provide exact quantitative results, it is shown that the cellular automata models capture essential dynamic patterns, which can be used to control the dynamics of networks and pathways. Cellular automata models of human diseases can help in the fight against cancer and HIV by simulating different strategies. Another field of application presented is the performance rate of network motifs with different topology, which might have evolutionary and biomedical importance. In Chapter 14, a model of microcirculation microcirculatory network in the form of a cellular automaton is proposed based on information about the anatomy and principles of functioning of the system. Its basic static and dynamic properties were investigated and a comparison with data from clinical investigations was carried out.

Dynamics of Social and Economic Systems. In Chapter 15, the properties of a cellular automaton that incorporates some assumptions from the Gaelic-Arvanitika model of language shifts and the findings on the dynamics of social impacts in the field of social psychology are introduced. A cellular automaton is defined and a set of simulations were carried out with it. Empirical data from recent sociolinguistic studies in Catalonia (a region in Southern Europe) were incorporated to run the automaton under different scenarios. It is also discussed how the social simulation based on cellular automata theory approach proves to be a useful tool for understanding language shifts. Chapter 16 provides an overview of cellular automata modelling approaches to socio-economic systems with emphasis on the spreading of innovations. The philosophy of bottom-up approaches of agent based models is outlined, and the typical set of cellular automata rules which have been proven successful during the past years in the field are described. As a specific example, there is a detailed presentation of cellular automata for the spreading of those types of technological innovations whose usage requires the so-called compatibility. It is the case, for instance, of the telecommunication technologies such as mobile phones, where a broad spectrum of devices is offered by the market with widely different technological levels. In Chapter 17, combining the feature of multi-agent system and complex network, a formal definition of cellular automata on networks is proposed and used to introduce a new artificial financial market modeling framework: Emergency-AFM. It includes classification and expression of information, uniform interfaces for investors' prediction and decision process, uniform interface for pricing mechanism, and analysis tools for time series. Chapter 18 introduces an approach for solving evolutionary flexible job-shop scheduling problem using cellular

automata. Genetic programming is applied in the algorithm; the rule tables undergo selection and crossover operations in the populations that follow.

Statistical Physics and Complexity. In Chapter 19, it is shown that elementary cellular automata with a single additive conserved quantity classify the density of the conserved quantity and that the same rules can show, when some stochastic boundary conditions are employed, a kind of nonequilibrium phase transition which is originally found in the asymmetric simple exclusion process. The probability distribution of patterns is calculated and the domain wall theory is applied to the elementary cellular automata. Diffusive behavior of the domain wall is discussed as well. Chapter 20 intends to show how simple cellular automata definitions allow construction of models reflecting physical properties of real systems, and to present how a complicated system evolution can be investigated with the help of cellular automata. In particular, the model of many two-level subsystems is discussed, some of which have been used and discussed extensively in physical models of solid state physics or quantum optics, but they also have been discussed as sociological or economical models. Chapter 21 describes the cellular automata simulation of two-layer Ising and Potts models. It was considered the isotropic ferromagnetic and symmetric case, using a two-layer square lattice with the periodic boundary condition. The Glauber method was used with checkerboard approach to update sites. In Chapter 22 it is considered how classic propositional logic and, in particular, propositional proof complexity can be combined with the study of cellular automata. The field of propositional proof complexity was born in the 1970s from two fields connected with computers: automated theorem proving and computational complexity theory. Here it is shown how propositional logic and techniques from propositional proof complexity can give a new proof of Richardson's Theorem, a famous theorem in this field. Also, some complexity results regarding cellular automata are considered and described, and the final section is devoted to a new proof system based on cellular automata. Chapter 23 presents an in silico model environment for the simulation of cardiac de- and repolarization and the three-dimensional potential pattern throughout the entire volume conductor. It is based on a cellular automaton and a bidomain-theory based source-field numerics. The in silico cardiac modelling solution presented enables various applications for the study of the nature of the ECG pattern in space and time. In Chapter 24, a study of viability of a visual processing model is presented. It has been defined by joining both cellular automata and spiking systems, that have important similarities and complement each other. Cellular automata make up a processing model for problem solving and spiking systems with address-event-representation give a solution for implementing a grid of neurons in hardware. Finally, Chapter 25 presents the design and implementation of CAOS, a domain-specific high-level programming language for the parallel simulation of extended cellular automata. CAOS allows scientists to specify complex simulations with limited programming skills and effort. Yet the CAOS compiler generates efficiently executable code that automatically harnesses the potential of contemporary multi-core processors, shared memory multiprocessors, workstation clusters and supercomputers. Both MPI (message passing interface) and OpenMP (an industry standard for shared memory programming) are used, either individually or in conjunction.

We hope that after reading different chapters of this book, we will succeed in bringing across what the scientific community is doing about the application of cellular automata

for modelling complex systems in a diversity of disciplines, and that the readers will find it interesting.

Lastly, we would like to thank all the authors for their excellent contributions in different areas of cellular automata modelling.

Alejandro Salcido
Instituto de Investigaciones Eléctricas
Cuernavaca,
Mexico

Part 1

Land Use and Population Dynamics

1

An Interactive Method to Dynamically Create Transition Rules in a Land-use Cellular Automata Model

Hasbani, J.-G., N. Wijesekara and D.J. Marceau
Department of Geomatics Engineering,
University of Calgary
Canada

1. Introduction

Cellular automata (CA) models are increasingly applied to simulate a wide range of spatio-temporal phenomena, including urban traffic (Sun and Wang, 2007), fire propagation (Ohgai *et al*., 2007), and insect infestation (Bone *et al*. 2006), but most importantly urban development (Almeida *et al*., 2008; Benenson and Torrens, 2004; Clarke *et al*., 1997; Santé *et al*., 2010; Shen *et al*., 2009; Van Vliet *et al*., 2009), and land-use changes (Ménard and Marceau, 2007; Moreno *et al*., 2010; Soares-Filho *et al*., 2002; Sui and Zeng, 2001). CA models are particularly suitable for land-use change modeling for several reasons. They are explicitly spatial and can be constrained in various ways to reflect local tendencies (Jenerette and Wu, 2001; Li and Yeh, 2000). It is also possible to specify for each simulated time step the quantity of land that should change from one land use to another (Jantz and Goetz, 2005). Information from a-spatial models, like a population growth model, can be integrated into the CA model to spatially allocate the land-use changes (White *et al*., 1997). A stochastic factor can also be included in the model to take into account some degree of unpredictability in the system (Moreno *et al*., 2009). As a consequence, CA models are often designed to test what-if scenarios and policies in urban and regional planning (Erlien *et al*., 2006; Jantz *et al*., 2003; Li and Yeh, 2004).

However, a challenge when implementing a CA model is its calibration. Calibration involves finding the parameters of the transition rules and the numerical values of these parameters so that the rules closely correspond to the dynamics of the system under investigation. This process is complicated due to the large number of combinations involved when several cell states, state transitions, parameters, and parameter values are being considered (Li and Yeh, 2002a; Shan *et al*., 2008). In addition, such combinations do not necessarily yield unique solutions (Verburg *et al*., 2004). Since there is no obvious way of finding which parameter should or should not be included in the model, the transition rules are often based on the modeler's intuitive understanding of the driving factors affecting the system (Wu, 2002).

Statistical techniques, such as logistic and multiple regressions (Fang *et al*., 2005; Sui and Zeng, 2001; Wu, 2002), principal component analysis (Li and Yeh, 2002a), and multivariate

analysis of variance (Lau and Kam, 2005) have been proposed for CA calibration. Computational intelligence techniques have also been tested, including artificial neural network (Li and Yeh, 2002b; Pijanowski *et al*., 2002), genetic algorithm (Shan *et al*., 2008), and data mining (Wang *et al*., 2010). Other methods involve the systematic testing of parameters (Jantz and Goetz, 2005; Jantz *et al*., 2003) and iterative calibration to achieve reasonable goodness-of-fit (Straatman *et al*., 2004). While these approaches might provide satisfactory simulation results, they often leave the modeler with little control on the mathematical equations used to determine the transition rules and the difficulty of understanding the geographical meaning of these rules (Verburg *et al*., 2004).

This paper describes a semi-automated, interactive method that was designed and implemented to dynamically create transition rules and calibrate a land-use CA model. The proposed method combines the benefits of conditional and mathematical rules and is adaptable in terms of number of land-use classes, and spatial and temporal scale of the input data. It allows the modeler to acquire information about the importance of the factors associated to historical land-use changes within the study area and to interactively select the parameter values required for the model calibration. A detailed description of the steps involved in the CA calibration is provided. The CA model is then used to answer the following questions: a) how sensitive is the model to the conditions involved in the calibration, including the cell size, neighborhood configuration, parameter values and external driving factors? b) what is the performance of the model, in terms of presence and location, in simulating land-use changes using the transition rules identified by the proposed calibration method?

2. Methodology

The study area is the dynamic eastern portion of the Elbow River watershed, located in southern Alberta, Canada, that covers an area of about 600 km^2 (Figure 1). The area is experiencing considerable pressure for land-use development due to the booming of the Alberta economy and its proximity to the City of Calgary, a fast growing city of one million inhabitants. About 5% of the watershed lies within the City of Calgary; 10% lies within the Tsuu T'ina nation, 20% within the municipal district of Rocky View, and the remaining 65% within the Kananaskis country. The study area is covered by about 48% of forest, 40% of agriculture and grassland, and 10% of built-up areas.

The historical land-use maps required for the CA calibration and validation were generated from Landsat Thematic Mapper imagery acquired during the summers of 1985, 1992, 1996, 2001, 2006 and 2010 at the spatial resolution of 30 m. Seven dominant classes were identified, namely evergreen, deciduous, agriculture, rangeland and parkland, built-up areas, water and clear-cut. Field verification was conducted for the years 2006 and 2010 and ancillary data along with expert knowledge were used to verify the classification results. A computer program was developed and applied to identify and correct minor spatial-temporal inconsistencies due to classification and georeference errors in the historical land-use maps.

A graph of the historical land-use trends reveals a decrease in the forested areas, a slight increase in parkland/rangeland, a sharper increase of built-up areas while agriculture slightly fluctuates, mostly from 2002 (Figure 2).

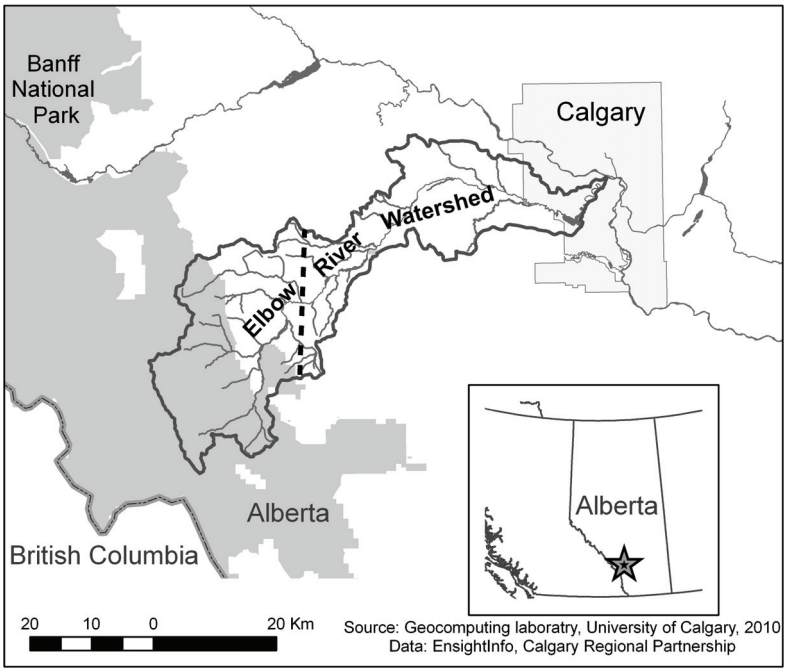


Fig. 1. Location of the study area; the dashed line represents the western limit of the study area

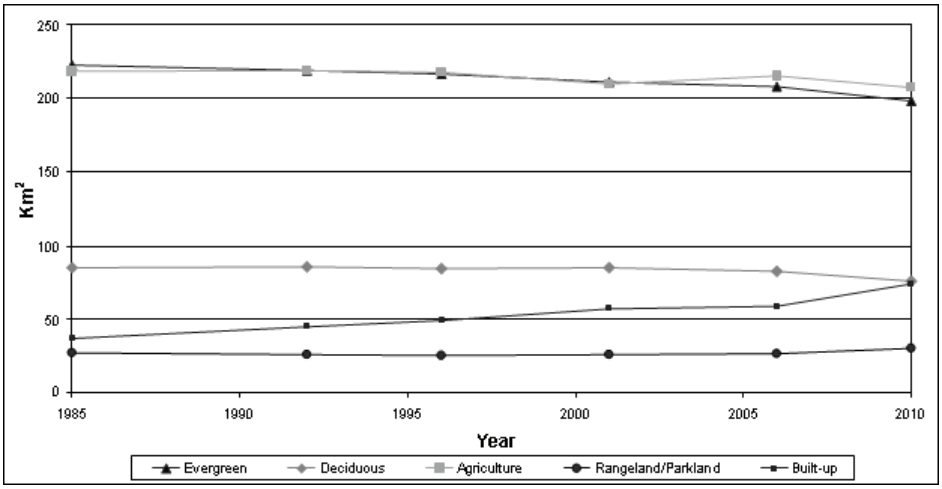


Fig. 2. Historical land-use trends in the study area

The historical land-use maps also indicate that a considerable amount of land-use transition occurred in the study area during the period considered (Table 1).

From	To	Land-use transition (%)	Total (%)
Evergreen	Agriculture	6.23	14.47
Deciduous		6.41	
Rangeland/Parkland		1.83	
Evergreen	Built-up	11.40	98.16
Deciduous		17.88	
Agriculture		65.97	
Rangeland/Parkland		2.91	
Agriculture	Rangeland /Parkland	43.52	43.52

Table 1. Amount of land-use transitions observed in the historical maps from 1985 to 2010. e.g. 14.47% is the percentage of agriculture increase in 2010 from the existing area of agriculture in 1985 and 6.23%, 6.41%, 1.83% are the contributing portions to this increase from each land-use transition to agriculture

2.1 Model implementation

The CA model was written in IDL version 6.3 from ITT Visual Information Solutions (ITTVIS, 2007). IDL is an array-oriented interpreted language based on optimized C routines. As a consequence, an operation on an array can be performed at a speed unreachable by a traditional for-loop going through each element of an array. IDL also offers the advantages of being a multiplatform language, of having internal functions dealing with spatial data, and of being linked to ENVI, a remote sensing image analysis software.

The model implementation includes three main steps: 1) the definition of the cell size, neighborhood configuration, and driving factors, 2) the transition rule extraction and the model calibration, and 3) the simulation procedure.

2.1.1 Cell size, neighborhood configuration, and driving factors selection

Several studies have shown that the cell size and neighborhood configuration have an impact on the outcomes of raster-based CA models and should not be arbitrarily chosen (Chen and Mynett, 2003; Kocabas and Dragicevic, 2006; Ménard and Marceau, 2005; Moreno *et al.*, 2009; Pan *et al.*, 2010; Samat, 2006; Benenson, 2007). To guide the selection of the cell size, an examination of the historical land-use maps was done, which revealed that most land-use changes were occurring over four or more contiguous pixels. To reduce computation time while maintaining the desired level of spatial details for the study, the land-use maps were resampled at the resolution of 60 and 100 m using the nearest neighbor algorithm available in ArcGIS 9.1 (ESRI, 2005).

The neighborhood was designed to approximate a circle around a center cell. This decision was made in order to reduce spatial distortions, when compared to an extended Moore neighborhood, as every cell located at a given distance from the center cell is considered in the neighborhood (Li and Yeh, 2002b). The modeler can choose the desired number and size of concentric neighborhood rings around a cell. The different rings are all exclusive; a cell can only be located in a single ring, and there is no gap between two rings (Figure 3). Within each ring, the influence of the neighboring cells on the central cell is constant but this

influence is different between rings. Consequently, the continuous distance function used in most CA models to represent the influence of neighborhood cells has been replaced by a discrete distance function. This approach has the main advantage of greatly simplifying the definition of the cells' influence as there is only one influence per ring. Moreover, these influences are dynamically found in the historical data and are not hard coded in the model, which allows the use of historical data at a different scale without changing the model.

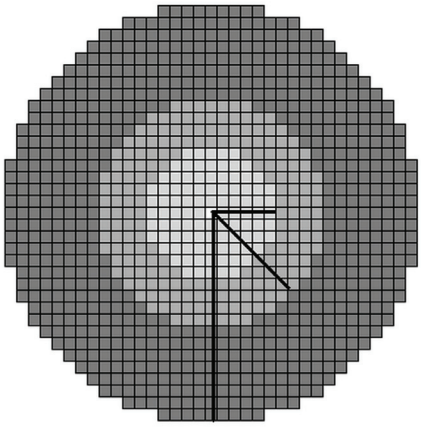


Fig. 3. Illustration of the neighborhood configuration used in the study corresponding to rings of 5, 9 and 17 cells

While testing all the possible combinations of cell size (60 m and 100 m) and neighborhood configuration was beyond the scope of this study, several combinations were tested to identify which ones provide the best simulation outcomes. Details regarding the sensitivity analysis that was conducted are provided in Section 2.1.3.

Land-use changes are complex spatial processes resulting from the interactions of socio-economic (e.g., population growth), biophysical (e.g., slope and soil quality), and geographic (e.g., proximity and accessibility to services) factors operating at different spatial and temporal scales (Liu and Phinn, 2003; Verburg *et al*., 2004). In this study, in addition to the influence of the cells located within local and extended neighborhoods as previously described, four external factors were considered as parameters in the transition rules, namely the distance to Calgary city center, the distance to a main road, the distance to a main river, and the ground slope. Such factors are commonly quoted in the literature as influencing land-use changes (Fang *et al*., 2005; Li and Yeh, 2002b; Pijanowski *et al*., 2002; Wu, 2002). The aforementioned distances were calculated for each cell and each historical year using the Euclidian distance tool available in ArcGIS 9.1 (ESRI, 2006). The resulting distance files were stored as raster images of the same resolution and extent as the land-use maps.

2.1.2 Rule extraction and model calibration

The transition rule extraction and the model calibration include the following steps (Figure 4). First, the set of historical land-use maps along with the maps corresponding to the driving factors are read and the number of cells of a particular state in the neighborhood of each central cell is computed. For each type of land-use change, all the cells that have

changed state in the historical land-use maps are identified. Frequency histograms are built to display the percentage of cells that have changed from one state to another when considering a particular driving factor and the cell state in the neighborhood. This provides quantitative information regarding the importance of each driving factor and neighborhood composition (i.e. state of the cells within the neighborhood) as being related to historical land-use changes within the study area. These histograms are interpreted by the modeler who identifies the ranges of values of each driving factor and neighborhood composition to be included in the conditional transition rules. This information is then automatically translated into mathematical transition rules.

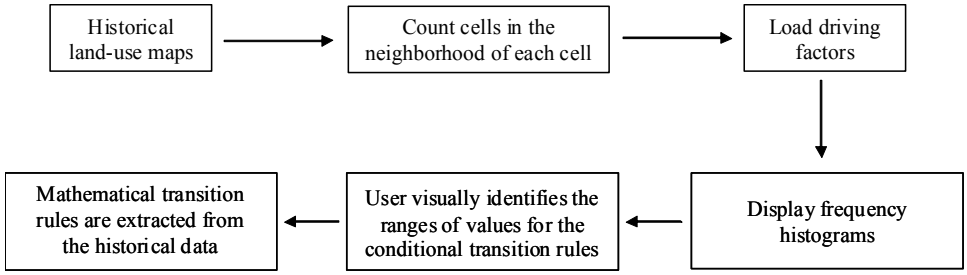


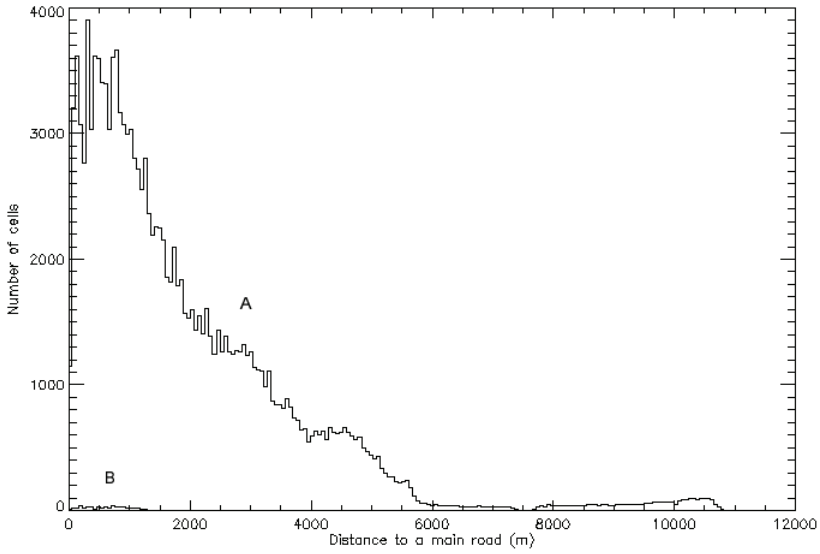
Fig. 4. Procedure for the extraction of the transition rules

Figure 5 provides an example of such a frequency histogram. The total number of Evergreen cells in the study area compared to the number of Evergreen cells that have changed to Built-up areas is first displayed to show the relative contribution of the later in the study area (Figure 5a). A detailed representation and analysis of the proportion of cells that have changed from Evergreen to Built-up areas when considering their distance to a main road (Figure 5b) reveals that 8% of these cells were located between 150 and 180 m of a main road while 98% of the cells were within 1250 m of a main road. At 1250 m, there is an inflexion point on the cumulative occurrence curve, expressing that this distance is critical for interpreting the influence of a main road on this land-use change. The further a cell was located from a main road, the less often it changed from Evergreen to Built-up area.

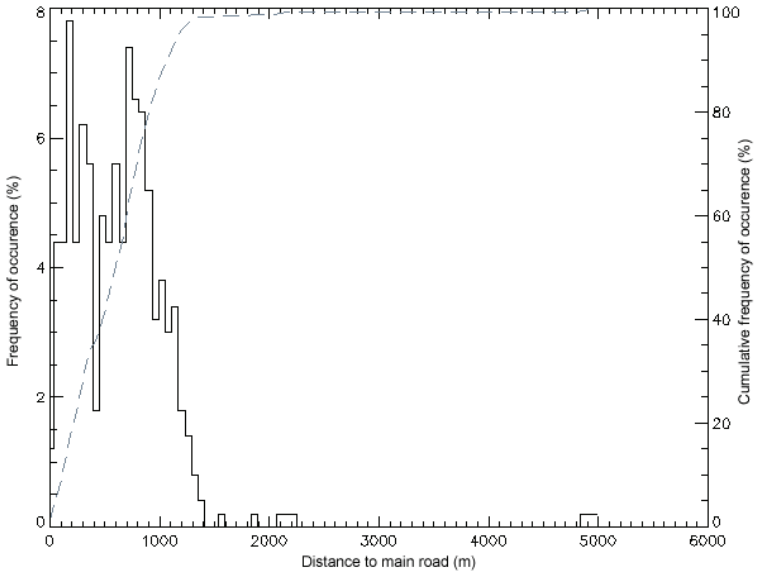
A graphical interface was designed to facilitate the interpretation of the frequency histograms and to allow a modeler to interactively select the ranges of values to be used for defining the conditional transition rules of the CA model (Figure 6). Each histogram can be displayed, allowing the modeler to change the bin size and to zoom in and out. By clicking on the histogram, the modeler identifies the ranges of values (minimum and maximum) for each neighborhood configuration, driving factor and cell state within that neighborhood. These values are stored in a table (Table 2) and further used to determine the conditional transition rules. An example of such a rule defined from Table 2 is:

If distance to a main road is between 0 and 427 m
and number of evergreen cells within the first neighborhood ring is between 0 and 17
and number of built-up cells within the second neighborhood ring is between 0 and 14
and number of agriculture cells within the third neighborhood ring is between 0 and 168
then the central Evergreen cell might change from Evergreen to Built-up area.

All possible transition rules are created by combining the identified ranges of values from the histograms.



a)



b)

Fig. 5. a) Frequency histogram comparing the total number of Evergreen cells located at a certain distance from a main road (A), with the number of Evergreen cells that have changed from Evergreen to Built-up areas when considering their distance to a main road (B); b) Frequency histogram displaying the percentage of cells that have changed from Evergreen to Built-up areas when considering their distance to a main road; the dashed curve represents the cumulative occurrence of the cells located at a certain distance from a main road

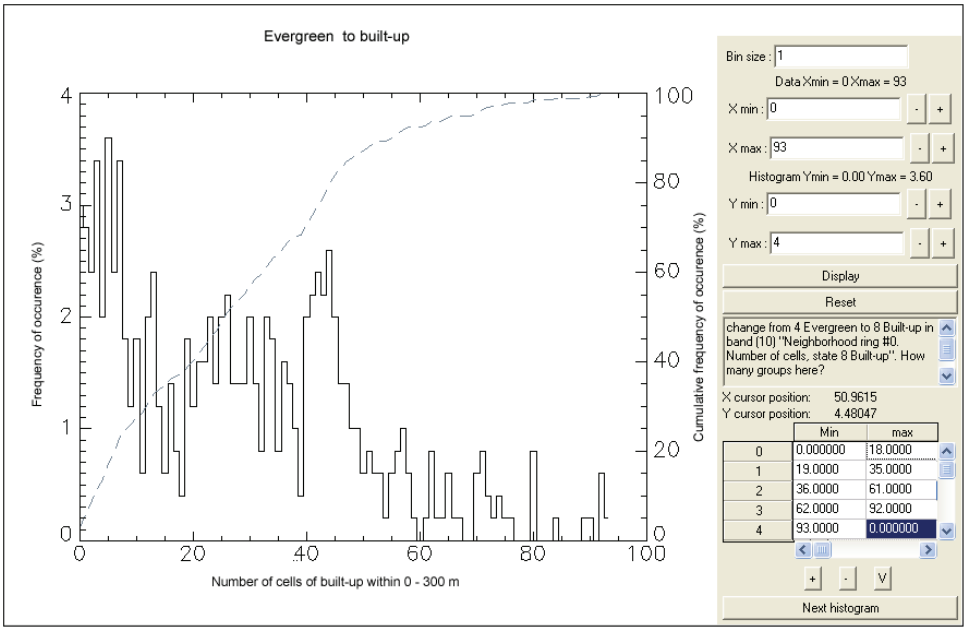


Fig. 6. Frequency histogram displaying the percentage of cells that have changed from Evergreen to Built-up when considering the number of Built-up cells within 300 m of these cells and graphical interface designed for the selection of the range of values to be considered in the conditional transition rules

Cell state	Distance to a main road (m)	Number of Evergreen cells located within the first neighborhood ring [0 to 300) m	Number of Built-up cells located within the second neighborhood ring [300 to 540) m	Number of Agriculture cells located within the third neighborhood ring [540 to 1020) m	…..
Evergreen	0 to 427	0 to 17	0 to 14	0 to 168	
	428 to 1408	18 to 50	15 to 59	169 to 258	
		51 to 74	60 to 92	259 to 377	

Table 2. Ranges of values identified from the frequency histogram to be used for determining the conditional transition rules

To convert the conditional rules into mathematical rules, the mean and standard deviation of the previously defined ranges of values are computed. These values become the coefficients of the parameters of the mathematical transition rules. In this model, the coefficients of each transition rule do not lead to a probability of change, but rather to a Resemblance Index (RI) that quantitatively describes the similarity between the neighborhood content of a cell at the time of the simulation and the neighborhood contents

that have been used to generate the values of the parameters of the transition rule. If they are very similar, it is likely that the cell should change state for the corresponding type of land use. RI is inspired by the Minimum Distance to Class Mean remote sensing image classification algorithm (Richards, 2006). This algorithm calculates the mean point in the parameter space for pixels of known classes and then assigns unknown pixels to the class that is arithmetically closest. It is computed for every transition rule using Equation 1.

$$RI = \sum_{i=1}^{m} \frac{\left| n_i - \bar{x}_i \right|}{\sigma_i} \tag{1}$$

where m is the number of layers (corresponding to the number of driving factors plus the number of land-use classes multiplied by the number of neighborhood rings), n_i is the value in layer i, $\bar{x}_i$ is the mean value for layer i in the transition rule and σ_i is the standard deviation for layer i in the transition rule. If the standard deviation is zero for layer i, then $\frac{\left| n_i - \bar{x}_i \right|}{\sigma_i} = 0$ if $n_i = \bar{x}_i$ or otherwise equals positive infinity. Accordingly, $RI \in \Re^+$ and the smaller RI is, the more similar is the cell neighborhood configuration to the ones used to define the transition rule. The mathematical rules offer greater flexibility compared to the conditional rules as they reflect significant values for each type of land-use change and are more adaptable to the neighborhood composition than the conditional rules identified from specific observations in the historical dataset.

Table 3 presents some values representing the coefficients of the conditional and mathematical rules, respectively for three neighborhood configurations. The Min and Max columns are associated to the conditional transition rule, while the Mean and Standard deviation columns are related to the mathematical transition rule. An example of a mathematical rule defined from these values is,

$$\begin{aligned}
RI(\text{rule1, Evergreen to Agriculture}) = {} & \frac{\left|\text{D. main road} - 259.38\right|}{173.05} + \frac{\left|\text{D. city center} - 6\,272.57\right|}{1\,568.91} + \\
& \frac{\left|\text{D. river} - 3\,465.61\right|}{310.77} + \frac{\left|\text{Ground slope} - 3.23\right|}{1.79} + \frac{\left|\text{N0_Water} - 0.17\right|}{0.44} + \frac{\left|\text{N0_Evergreen} - 10.71\right|}{5.25} + \\
& \frac{\left|\text{N0_Deciduous} - 3.13\right|}{3.42} + \frac{\left|\text{N0_Agriculture} - 81.84\right|}{5.75} + \frac{\left|\text{N0_Rangeland / Parkland} - 0.04\right|}{0.29} + \\
& \frac{\left|\text{N0_Built - up} - 0.08\right|}{0.35} + \left\{0 \text{ if N0_Clear - cut} = 0; \infty \text{ otherwise}\right\} + \frac{\left|\text{N1_Water} - 0.44\right|}{0.86} + \\
& \frac{\left|\text{N1_Evergreen} - 19.24\right|}{10.0} + \frac{\left|\text{N1_Deciduous} - 4.75\right|}{3.9} + \frac{\left|\text{N1_Agriculture} - 170.93\right|}{11.51} + \\
& \frac{\left|\text{N1_Rangeland / Parkland} - 0.28\right|}{0.86} + \frac{\left|\text{N1_Built - up} - 0.33\right|}{0.63} + \left\{0 \text{ if N1_Clear - cut} = 0; \infty \text{ otherwise}\right\} +
\end{aligned}$$

$$\frac{|N2\_Water - 1.46|}{1.94} + \frac{|N2\_Evergreen - 86.53|}{43.49} + \frac{|N2\_Deciduous - 14.53|}{12.15} +$$
$$\frac{|N2\_Agriculture - 573.8|}{44.90} + \frac{|N2\_Rangeland\ /\ Parkland - 1.51|}{3.76} + \frac{|N2\_Built - up - 2.13|}{4.15} +$$
$$\frac{|N2\_Clear - cut - 0.02|}{0.14}$$

where Nx_LandUse is the number of cells of the corresponding land use within neighborhood rings of a cell. All the transition rules are stored in a file, so multiple simulations can be performed without re-calibrating the model.

Neighborhood ring	Layer	Min	Max	Mean	StdDev
Cell attributes	Cell state	Evergreen			
	Dist. to main road	0.0	780	259.38	173.05
	Dist. to city center	2473.86	8696.48	6272.57	1568.91
	Dist. to river	2979.53	4048.11	3465.61	310.77
	Ground Slope	0.0	7.29	3.23	1.79
[0-300) m	Nb cells state Water	0.0	2.0	0.18	0.44
	Nb cells state Evergreen	0.0	19.0	10.71	5.25
	Nb cells state Deciduous	0.00	11.00	3.13	3.43
	Nb cells state Agriculture	66.00	93.00	81.84	5.76
	Nb cells state Rangeland/Parkland	0.00	2.00	0.04	0.30
	Nb cells state Built-up	0.00	2.00	0.09	0.36
	Nb cells state Clear-cut	0.00	0.00	0.00	0.00
[300-540) m	Nb cells state Water	0.00	3.00	0.44	0.87
	Nb cells state Evergreen	0.00	32.00	19.24	10.00
	Nb cells state Deciduous	0.00	16.00	4.76	3.91
	Nb cells state Agriculture	151.00	192.00	170.93	11.52
	Nb cells state Rangeland/Parkland	0.00	4.00	0.29	0.87
	Nb cells state Built-up	0.00	3.00	0.33	0.64
	Nb cells state Clear-cut	0.00	0.00	0.00	0.00
[540-1020) m	Nb cells state Water	0.00	6.00	1.47	1.95
	Nb cells state Evergreen	2.00	205.00	86.53	43.49
	Nb cells state Deciduous	1.00	51.00	14.53	12.16
	Nb cells state Agriculture	430.00	656.00	573.80	44.91
	Nb cells state Rangeland/Parkland	0.00	14.00	1.51	3.76
	Nb cells state Built-up	0.00	24.00	2.13	4.15
	Nb cells state Clear-cut	0.00	1.00	0.02	0.15

Table 3. Example of data used to establish the conditional and mathematical transition rules

2.1.3 Simulation procedure

The simulation procedure includes these main steps.

- The mathematical transition rules previously defined and a land-use map corresponding to the beginning of the simulation are read.
- For each time step, the neighborhood configuration of every cell is read and the level of correspondence with the parameters of the transition rules is computed.
- With respect to the user-specified constraints and to the influence of each rule, the cells that change state do it according to the rule having the highest level of correspondence.
- To decide which cell should be associated to each type of land-use change, the model recursively sorts the type of land-use changes and for each of them selects the cell having the smallest RI value. Once the required number of cells associated to each type of land-use change is met or when no more cells can be assigned, the model writes the new land-use map and updates the statistics that correspond to the percentage of cells associated to each rule and each type of change.
- If the numbers of cells associated to each rule and each type of land-use change is different than the numbers found from the historical data and previous time steps, a correction is applied at the next time step. For example, if 200 cells are to change from Agriculture to Built-up area but only 150 of them can according to their neighborhood configuration, 50 additional cells will be set to change at the next time step.

Table 4 lists the land-use transitions that were considered during the simulations.

From	To
Evergreen	Agriculture
Deciduous	Agriculture
Evergreen	Built-up
Deciduous	Built-up
Agriculture	Built-up
Rangeland/Parkland	Built-up
Rangeland/Parkland	Agriculture
Agriculture	Rangeland/Parkland

Table 4. Land-use transitions considered during the simulations

Two sets of simulations were run. The first set was to test the model under various conditions. A sensitivity analysis to the cell size and neighborhood configuration was first carried out, followed by a sensitivity of the model to different ranges of values selected from the frequency histograms. Several ways of grouping the data values extracted from the histograms were tested for the calibration of the CA model: 1) the most dominant ranges of values ignoring flat areas, 2) the most dominant range of values concentrated around the mode, 3) the most dominant range of values dispersed away from the mode, and 4) the whole range of values of the histogram. Finally, to assess the importance of the selection of the external driving factors, simulations were conducted using four factors and different combinations of only three external factors.

In each case, the CA was calibrated using the land-use maps of 1985 to 2001, simulations were performed from 2001 to 2006, and the simulated map of 2006 was compared to the reference map of the same year using two similarity measures based on Kappa coefficients. The first measure is the standard Kappa coefficient, which expresses the percentage of

agreement between two maps including both quantity and location information (Hagen, 2003; Pan *et al*., 2010; Visser and de Nijs, 2006). Three statistics were calculated, respectively referred to as Kappa, Kloc and Khisto. Khisto measures the quantitative similarity between two compared maps, while Kloc measures the similarity of the spatial allocation of categories between the maps. Kappa represents the general level of spatial agreement between two maps and is the product of Kloc and Khisto.

A drawback of the standard Kappa statistics is that they tend to over-estimate the agreement between a simulated map and a reference map because they do not take into account the percentage of cells that do not change state during the simulation period. In addition, they rely on a stochastic model of random allocation based on the sizes of the classes being compared to express the expected agreement. When simulating with a CA model, land-use allocation is not totally random since it depends on the initial conditions of the simulation. To compensate for these limitations, Van Vliet *et al*. (2010) introduced a coefficient of agreement called Kappa simulation that applies a more appropriate stochastic model of random allocation of class transitions that takes into account the information contained in the initial land-use map and the proportion of cells that does not change state over the simulation period. Three statistics were again calculated: Ksimulation that expresses the agreement between the simulated land-use map and the reference map, Ktransition that captures the agreement in terms of quantity of land-use transitions, and Ktransloc that measures the agreement between the two maps in terms of location of transition. Values of these coefficients vary from -1 to 1, the former value indicating a perfect disagreement between the two maps compared while the later indicating a perfect agreement. The standard and Kappa simulation coefficients were calculated using the Map Comparison Kit developed by the Research Institute for Knowledge System (RIKS BV, 2010).

To carry out a validation test, a simulation was conducted using the best combination of conditions described above from 2001 to 2006 and to 2010. A comparison was performed between the simulated maps and the reference maps for the years 2006 and 2010. An additional simulation was conducted from 1985 to 2010 to illustrate how the simulated land uses change over the whole period of time compared to the changes observed in the reference maps.

In all these simulations, a local constraint was applied to forbid built-up cells within the Tsuu T'ina nation. For the validation test where simulations were conducted from 2001 to 2010 and from 1985 to 2010, and where the selection of external driving factors was tested, a global constraint was also applied to restrict the number of built-up cells at each iteration based on an average estimated from the historical population trends.

3. Results

3.1 Sensitivity analyses

Table 5 presents the coefficients of agreement obtained when using a cell size of 60 m and 100 m, respectively. As expected, the values of the standard Kappa statistics tend to be high. They are also very similar and do not allow a discrimination among the results. However, the values of Ksim, Ktransloc and Ktransition all reveal that the simulation results obtained with 60 m are in higher agreement with the reference map than the results achieved using a cell size of 100 m.

The Ksim coefficient also shows that the choice of neighborhood configuration affects the simulation outcomes. Using only two rings in the neighborhood definition considerably reduces the performance of the model, while the best outcome is achieved when using three rings of respectively 5, 9 and 17 cells. This indicates that an extended neighborhood that covers a distance up to 1020 m is more appropriate in this study area to capture the zone of influence on central cells.

Cell size	Standard Kappa	Kloc	Khisto	Ksim	Ktransloc	Ktransition
60 m	0.853	0.875	0.975	0.047	0.085	0.551
100 m	0.850	0.873	0.974	0.043	0.078	0.546

Table 5. Kappa coefficients of agreement obtained when using a cell size of 60 m and 100 m

Neighborhood Configuration	Standard Kappa	Kappa simulation
3-5	0.845	0.015
3-5-15	0.850	0.037
5-9-14	0.852	0.044
5-9-15	0.852	0.045
5-9-16	0.853	0.046
5-9-17	0.853	0.047
5-12-17	0.852	0.045
6-9-15	0.849	0.031
6-14-18	0.852	0.043
6-14-19	0.852	0.045
7-10-15	0.852	0.042
7-10-16	0.852	0.044
7-10-17	0.852	0.044
7-13-17	0.850	0.034
7-14-18	0.852	0.043
7-14-19	0.852	0.044
7-14-20	0.852	0.045
7-15-19	0.852	0.043
8-12	0.847	0.024
8-15-19	0.852	0.042

Table 6. Kappa coefficients of agreement obtained when using different neighborhood configurations

The way ranges of values are selected from the frequency histograms to build the transition rules also affects the simulation outcomes (Table 7). The best results are achieved when the values are concentrated around the mode (Ksim = 0.069) compared to progressively more dispersed ranges of values (Ksim = 0.047 and 0.045). The worse result is achieved when a single range of values covering the whole histogram is selected (Ksim = 0.041).

Selection of parameter values	Kappa	KLocation	KHisto	Kappa simulation	KTransLoc	KTransition
Most dominant ranges of values	0.853	0.875	0.975	0.047	0.085	0.551
Values dispersed from the mode	0.853	0.875	0.975	0.045	0.081	0.551
Values concentrated around the mode	0.857	0.879	0.975	0.069	0.126	0.551
One group of values	0.852	0.874	0.975	0.041	0.074	0.551

Table 7. Kappa coefficients of agreement obtained when using different grouping of values from the frequency histograms for the definition of the transition rules

Simulation outcomes are also influenced by the number and selection of external driving factors (Table 8). Using four factors generates the highest agreement with the reference map both in terms of overall agreement (0.058) and location (0.140). The best combination of three factors includes distance to main road, distance to city center and distance to river, which were expected to play a major role in the increase of built-up areas. Ground slope also appears to be an important factor as revealed by the coefficients of agreement that are slightly lower than the ones obtained with the previous three factors. It can be observed that the amount of land-use transitions remains the same with the different combinations of factors; however, their spatial distribution changes as indicated by Ktransloc.

Factor selection	Standard Kappa	Kloc	Khisto	Ksim	KtransLoc	Ktrans
Dist. to main road Dist. to city center Dist. to river Ground Slope	0.866	0.876	0.989	0.058	0.140	0.411
Dist. to main road Dist. to city center Dist. to river	0.864	0.874	0.989	0.042	0.102	0.411
Dist. to main road Dist. to city center Ground Slope	0.864	0.873	0.989	0.038	0.094	0.411
Dist. to main road Dist. to river Ground Slope	0.864	0.874	0.989	0.041	0.100	0.411
Dist. to city center Dist. to river Ground Slope	0.864	0.874	0.989	0.041	0.101	0.411

Table 8. Kappa coefficients of agreement obtained when using four external driving factors compared to the combinations of only three factors

3.2 Results obtained with the best combination of conditions

The Kappa coefficients of agreement obtained when running simulations from 2001 to 2006 and to 2010 using the best combination of conditions are presented in Table 10. The agreement is higher for the year 2006 compared to the year 2010. A more detailed analysis of the results provided by the per-class Kappa simulation coefficients for the years 2006 and 2010 indicates that in terms of number of transition, the model achieves a relatively good agreement with the reference maps (values between 0.371 and 0.541), except for the class built-up where the values are slightly over 0.2 (Tables 11 and 12). The values obtained for Ktransloc are lower than those obtained for Ktransition indicating that the model is better at allocating the right amount of transition rather than their location.

	Year	
	2006	**2010**
Standard Kappa	0.869	0.782
Kappa simulation	0.075	0.057

Table 10. Overall Kappa coefficients of agreement obtained when running simulation from 2001 to 2010 and comparing the results with the reference maps of 2006 and 2010

	Built-up	**Rangeland/ parkland**	**Agriculture**	**Deciduous**	**Evergreen**
Standard Kappa	0.816	0.669	0.860	0.906	0.947
Kloc	0.817	0.684	0.865	0.917	0.948
Khisto	0.999	0.978	0.994	0.988	0.999
Kappa simulation	0.009	0.140	0.096	0.084	0.065
KtransLoc	0.045	0.259	0.178	0.165	0.174
Ktransition	0.211	0.541	0.536	0.508	0.376

Table 11. Per-class Kappa coefficients of agreement obtained when running simulation from 2001 to 2006 and comparing the simulated map of 2006 with the reference map of 2006

	Built-up	**Rangeland/ parkland**	**Agriculture**	**Deciduous**	**Evergreen**
Standard Kappa	0.706	0.537	0.779	0.822	0.905
Kloc	0.793	0.553	0.806	0.834	0.930
Khisto	0.890	0.971	0.967	0.986	0.973
Kappa simulation	0.011	0.095	0.078	0.079	0.046
KtransLoc	0.054	0.198	0.148	0.171	0.123
Ktransition	0.204	0.483	0.530	0.460	0.371

Table 12. Per-class Kappa coefficients of agreement obtained when running simulation from 2001 to 2010 and comparing the simulated map of 2010 with the reference map of 2010

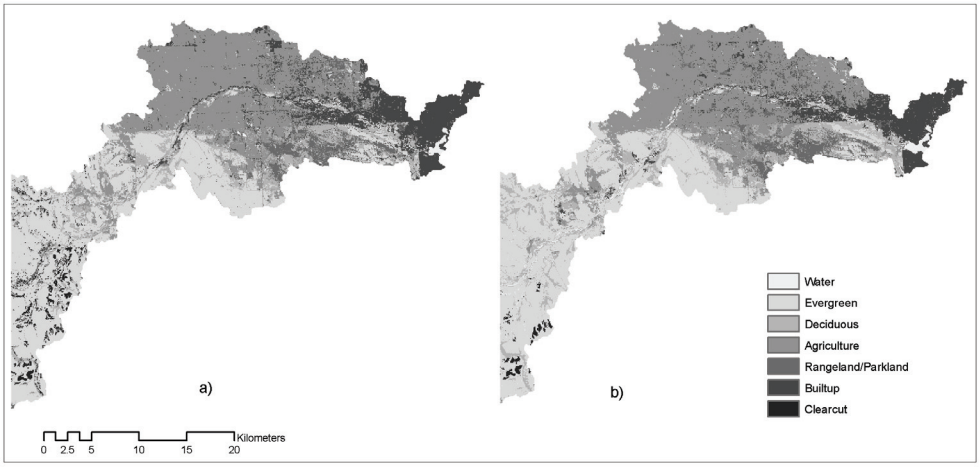


Fig. 7. Comparison of the reference map of 2010 (a) with the simulated map of the same year (b)

A visual comparison of the 2010 reference and simulated maps shows an under-estimation of built-up cells by the model. This is mainly due to the considerable urban growth that occurred during the period 2006-2010 that is not accounted for in the global constraint of the model. The simulation of agricultural areas is affected by the above since most built-up cells in the reference map are located within the agricultural areas.

Simulations run from 1985 to 2010 provide additional details and reveal that the CA model under-estimates the quantity of built-up areas (difference of 4.37% in 2010) and over-estimates agriculture (difference of 5.73% in 2010) (Table 9). The proportion of rangeland/parkland simulated by the model does not differ greatly from the proportion observed in the reference maps. The under-estimation of deciduous can be explained by the fact that the CA model does not simulate the transition to this class.

Land-use	1985	1992	1996	2001	2006	2010
Evergreen	36.87	35.80(-0.47)	35.47(-0.40)	35.15(0.19)	34.61(0.18)	34.19(1.34)
Deciduous	14.08	12.67(-1.50)	12.19(-1.85)	11.68(-2.45)	10.93(-2.81)	10.33(-2.29)
Agriculture	36.16	37.56(1.29)	38.12(2.03)	38.65(3.91)	39.43(3.82)	40.06(5.73)
Rangeland /Parkland	4.48	5.07(0.75)	5.04(0.77)	4.99(0.62)	5.14(0.75)	5.26(0.25)
Built-up	6.12	6.61(-0.80)	6.90(-1.31)	7.25(-2.19)	7.60(-2.18)	7.88(-4.37)

Table 9. Percentage of the study area covered by the main land uses in the simulation results from 1985 to 2010. The variation with the original land-use maps is shown in parentheses; a positive value indicates an over-estimation by the model while a negative value indicates the opposite.

The fact that values of Kappa simulation and Ktransloc are sometimes relatively low might be explained by the large number of land-use transitions considered in this study and the difficulty of capturing the dynamics specific to each type of transition. To obtain a preliminary assessment of how the model would perform with a reduced number of land-use classes and transitions, a simulation was run with aggregated land-use maps in which

the number of land-use classes was reduced to five (water, forest, agriculture, built-up, and Tsuu T'ina nation) and only four transitions were considered, namely forest to agriculture, forest to built-up, agriculture to forest, and agriculture to built-up. The model was calibrated over the period 1985-2001 and the simulation was run from 1985 to 2006 using three external driving factors (distance to Calgary city center, distance to a main road, distance to the main river). Higher values were obtained for the three Kappa simulation statistics calculated (Table 14), confirming that the simulation results could be improved by either reducing the number of land-use transitions in the model or improving the rules for some of the land-use transitions considered.

	1992	1996	2001	2006
Kappa	0.947	0.941	0.929	0.917
KLocation	0.951	0.947	0.938	0.939
KHisto	0.996	0.994	0.991	0.976
Kappa Simulation	0.304	0.262	0.251	0.227
KTransLoc	0.429	0.377	0.336	0.341
KTransition	0.709	0.695	0.747	0.665

Table 14. Overall Kappa coefficients of agreement obtained when running simulation with a reduced number of land-use transitions.

4. Conclusion

While the potential of CA models is increasingly acknowledged for land-use change studies, their calibration and the evaluation of their performance remains a challenge. In this research, we developed a calibration method that allows the modeler to interactively obtain information about historical land-use changes and the factors associated to these changes in order to automatically derive conditional and mathematical transition rules. When testing the applicability of this model in a study area in southern Alberta, sensitivity analyses were conducted to evaluate the influence of various conditions involved in the calibration of the model, including the cell size, the neighborhood configuration, the selection of the parameter values, and the number of driving factors. These analyses indicate that the simulation outcomes are affected by the selection of these conditions and that there exist no method to *a priori* identify the most adequate combination.

Sensitivity of raster-based CA models to cell size and neighborhood configuration has been recognized by several authors over the last years. One approach to overcome this sensitivity to scale is the implementation of object-based CA models with the inclusion of a dynamic neighborhood as proposed by Moreno *et al*. (2009, 2010) and others (Hamman *et al*., 2007). While such models are computationally intensive and the handling of the topology cumbersome, they appear as a promising approach to better capture the meaningful entities composing a landscape along with their evolution.

The calibration technique described in this paper provides useful insights regarding the number and choice of external driving factors that should be considered in the calibration. When taking into account external factors in addition to the influence of the cells within extended neighborhoods, the number of possible combinations of factors becomes too high to be thoroughly evaluated by a simple sensitivity analysis. Other approaches based on data

mining techniques might be useful in this context to guide the selection of driving factors for the calibration of the model (Wang *et al*., in press).
Kappa simulation is a recently proposed coefficient of agreement specifically adapted to the context of evaluating the performance of a CA model (Van Vliet *et al*., 2010). While additional studies are needed to fully assess the interpretation potential of this coefficient, it appears very useful in this study to capture differences in simulation results when the standard Kappa was not sensitive enough to provide discrimination. In particular, it indicates that the CA model generates a relatively high agreement in terms of amount of land-use transitions, while the agreement in terms of location is lower. The fact that the values of the coefficients increase when reducing the number of land-use transitions considered in the model also reveals that additional external driving factors might be necessary to fully capture the dynamics of the study area.
When interpreting the values of these coefficients however, we must keep in mind that they inform on the agreement between two maps on a cell-by-cell basis, without considering a slight displacement that might occur among the cells being compared. In addition, the comparison is performed between two possible states of the area being studied, respectively generated by the model and from observations acquired at a specific moment in time. These two states might differ, which does not necessarily imply that the simulated outcomes are 'wrong'.

5. Acknowledgments

This project was initiated in collaboration with the Calgary Regional Partnership (CRP) who provided financial support to J.-G. Hasbani. We thank Cheng Zhang from the University of Calgary for producing the historical land-use maps used in the project and Jasper Van Vliet for the stimulating discussions regarding Kappa simulation. Additional funding was provided by a NSERC Discovery grant awarded to D. Marceau and by GEOIDE, the Canadian Network of Centers of Excellence in Geomatics.

6. References

Almeida, C. M., Gleriani, J. M., Castejon, E. F., & Soares-Filho, B. S., 2008. Using neural networks and cellular automata for modelling intra-urban land-use dynamics. *International Journal of Geographical Information Science* 22(9): 943-963.

Benenson, I., & Torrens, P., 2004. *Geosimulation: Automata-based modeling of urban phenomena*, Wiley and Sons.

Benenson, I., 2007. Warning! The scale of land-use CA is changing! *Computers, Environment and Urban Systems* 31(2): 107-113.

Bone, C., Dragicevic, S., & Roberts, A., 2006. A fuzzy-constrained cellular automata model of forest insect infestations. *Ecological Modelling* 192: 107-125.

Calgary Economic Development, 2010, http://www.calgaryeconomicdevelopment.com/liveWorkPlay/Live/demographics.cfm

Chen, Q., & Mynett, A. E., 2003. Effects of cell size and configuration in cellular automata based prey-predator modelling. *Simulation Modelling Practice and Theory* 11(7-8): 609-625.

Clarke, K. C., Hoppen, S., & Gaydos, L., 1997. A self-modifying cellular automaton model of historical urbanization in the San Francisco Bay area. *Environment and Planning B* 24(2): 247-261.

Erlien, C. M., Tuttle, J. P., McCleary, A. L., Mena, C.F., & Walsh, S.J., 2006. Complexity theory and spatial simulations of land use/land cover dynamics: the use of what if scenarios for education, land management, and decision-making. *Geocarto International* 21(4): 67-74.

ESRI, 2006. ArcGIS 9.1 Users' manual. ESRI, Redlands, California.

Fang, S., Gertner, G. Z., Sun, Z., & Anderson, A.A., 2005. The impact of interactions in spatial simulation of the dynamics of urban sprawl. *Landscape and Urban Planning* 73: 294-306.

Hagen, A., 2003. Fuzzy set approach to assessing similarity of categorical maps. *International Journal of Geographical Information Science* 17(3): 235-249.

Hammam, Y., A. Moore, & Whigham, P. (2007). The dynamic geometry of geographical vector agents. *Computers, Environment and Urban Systems* 31: 502-519.

Jantz, C. A., & Goetz, S. J., 2005. Analysis of scale dependencies in an urban land-use-change model. *International Journal of Geographical Information Science* 19(2): 217-241.

Jantz, C. A., Goetz, S. J., & Shelley, M.K., 2003. Using the SLEUTH urban growth model to simulate the impacts of future policy scenarios on urban land use in the Baltimore-Washington metropolitan area. *Environment and Planning B* 30: 251-271.

Jenerette, D. G., & Wu, J., 2001. Analysis and simulation of land-use change in the central Arizona – Phoenix region, USA. *Landscape Ecology* 16:611-626.

Kocabas, V., & Dragicevic, S., 2006. Assessing cellular automata model behaviour using a sensitivity analysis approach. *Computers, Environment and Urban Systems* 30: 921-953.

Lau, K. H., & Kam, B. H., 2005. A cellular automata model for urban land-use simulation. *Environment and Planning B* 32: 247-263.

Li, X., & Yeh, A. G.-O., 2000. Modelling sustainable urban development by the integration of constrained cellular automata and GIS. *International Journal of Geographical Information Science* 14(2): 131-152.

Li, X., & Yeh, A. G.-O., 2002a. Integration of principal components analysis and cellular automata for spatial decision making and urban simulation. *Science in China*, 45(6): 521-529.

Li, X., & Yeh, A. G.-O., 2002b. Neural-network-based cellular automata for simulating multiple land use changes using GIS. *International Journal of Geographical Information Science* 16(4): 323-343.

Li, X., & Yeh, A. G.-O., 2004. Data mining of cellular automata's transition rules. *International Journal of Geographical Information Science* 18(8): 723-744.

Liu, Y., & Phinn, S. R., 2003. Modelling urban development with cellular automata incorporating fuzzy-set approaches. *Computers, Environment and Urban Systems* 27: 637-658.

Ménard, A., & Marceau, D. J., 2005. Exploration of spatial scale sensitivity in geographic cellular automata. *Environment and Planning B* 32: 693-714.

Ménard, A., & Marceau, D. J., 2007. Simulating the impact of forest management scenarios in an agricultural landscape of southern Quebec, Canada, using a geographic cellular automaton. *Landscape and Urban Planning* 79(3-4): 253-265.

Moreno, N., Wang, F., & Marceau, D.J., 2009. Implementation of a dynamic neighborhood in a land-use vector-based cellular automata model. *Computers, Environment and Urban Systems*, 33: 44-54.

Moreno, N., Wang, F., & D.J. Marceau, 2010. A geographic object-based approach in cellular automata modeling. *Photogrammetric Engineering and Remote Sensing* 76(2): 183-191.

Ohgai, A., Gohnai, Y., & Watanabe, K., 2007. Cellular automata modeling of fire spread in built-up areas: A tool to aid community-based planning for disaster mitigation. *Computers, Environment and Urban Systems* 31(4): 441-460.

Pan, Y., Roth, A., Yu, Z., & Doluschitz, R., 2010. The impact of variation in scale on the behavior of a cellular automata used for land-use change modeling. *Computers, Environment and Urban Systems* 34: 400-408.

Pijanowski, B. C., Brown, D.G., Shellito, B.A., & Manik, G.A. 2002. Using neural networks and GIS to forecast land use changes: a land transformation model. *Computers, Environment and Urban Systems* 26: 553-575.

Richards, J. A., 2006. *Remote sensing digital image analysis: An introduction*, Springer, 439 p.

Research Institute for Knowledge System (RIKS BV), 2010, http://www.riks.nl/mck

Samat, N., 2006. Characterizing the scale sensitivity of the cellular automata simulated urban growth, A case study of the Seberang Perai Region, Penang State, Malaysia Computers, *Environment and Urban Systems* 30(6): 905-920.

Santé, I., Garcia, A. M., Miranda, D., & Crecente, R., 2010. Cellular automata models for the simulation of real-world urban processes: a review and analysis. *Landscape and Urban Planning* 96: 108-122.

Shan, J., Alkheder, S., & Wang, J., 2008. Genetic algorithms for the calibration of cellular automata urban growth modeling. *Photogrammetric Engineering and Remote Sensing* 74(10): 1267-1277.

Shen, Z., Kawakami, M., & Kawamura, I., 2009. Geosimulation model using geographic automata for simulating land-use patterns in urban partitions. *Environment and Planning B* 36: 802-823.

Soares-Filho, B. S., Cerqueira, G. C., & Pennachin, C.L., 2002. DINAMICA: a stochastic cellular automata model designed to simulate the landscape dynamics in an Amazonian colonization frontier. *Ecological Modelling* 154: 217-235.

Straatman, B., White, R., & Engelen, G., 2004. Towards an automatic calibration procedure for constrained cellular automata. *Computers, Environment and Urban Systems* 28: 149-170.

Sui, D. Z., & Zeng, H., 2001. Modeling the dynamics of landscape structure in Asia's emerging desakota regions, a case study in Shenzhen. *Landscape and Urban Planning* 53(1-4): 37-52.

Sun, T., & Wang, J., 2007. A traffic cellular automata model based on road network grids and its spatial and temporal resolution's influences on simulation. *Simulation Modelling Practice and Theory* 15: 864-878.

Van Vliet, J., White, R., & Dragicevic, S., 2009. Modeling urban growth using a variable grid cellular automata. *Computers, Environment and Urban Systems* 33: 35-43.

Van Vliet, J., Bregt, A. K., & Hagen-Zanker, A., 2010. Revisiting Kappa to account for change in the accuracy assessment of land-use change models. Submitted to *Ecological Modelling*.

Verburg, P. H., de Nijs, T. C. M., Van Eck, J.R., Visser, H., & de Jong, K., 2004. A method to analyse neighbourhood characteristics of land use patterns. *Computers, Environment and Urban Systems* 28(6): 667-690.

Visser, H., & de Nijs, T., 2006. The Map Comparison Kit. *Environmental Modelling and Software* 21(3): 346-358.

Wang, F., Hasbani, J-G., Wang, X., & Marceau, D. J., 2011. Identifying dominant factors for the calibration of a land-use cellular automata model using Rough Set theory. *Computers, Environment and Urban Systems*, in press.

White, R., Engelen, G., & Uljee, I., 1997. The use of constrained cellular automata for high-resolution modelling of urban land-use dynamics. *Environment and Planning B* 25: 323-343.

Wu, F., 2002. Calibration of stochastic cellular automata, the application to rural-urban land conversions. *International Journal of Geographical Information Science* 16(8): 795-818.

2

Cellular-Automata-Based Simulation of the Settlement Development in Vienna

Reinhard Koenig[1] and Daniela Mueller[2]
[1]*Bauhaus-University Weimar, Faculty of Architecture, Chair Computer Science in Architecture,*
[2]*TU Vienna, Department of Spatial Development, Infrastructure & Environmental Planning, Centre of Regional Science*
[1]*Germany*
[2]*Austria*

1. Introduction

The structure and development of cities can be seen and evaluated from different points of view. By replicating the growth or shrinkage of a city using historical maps depicting different time states, we can obtain momentary snapshots of the dynamic mechanisms of the city. An examination of how these snapshots change over the course of time and a comparison of the different static time states reveals the various interdependencies of population density, technical infrastructure and the availability of public transport facilities. Urban infrastructure and facilities are not distributed evenly across the city – rather they are subject to different patterns and speeds of spread over the course of time and follow different spatial and temporal regularities.

The reasons and underlying processes that cause the transition from one state to another result from the same recurring but varyingly pronounced hidden forces and their complex interactions. Such forces encompass a variety of economic, social, cultural and ecological conditions whose respective weighting defines the development of a city in general. Urban development is, however, not solely a product of the different spatial distribution of economic, legal or social indicators but also of the distribution of infrastructure. But to what extent is the development of a city affected by the changing provision of infrastructure?

As Lichtenberger (1986, p. 154) already notes, urban structures have often been characterized by the development of technical and socio-cultural infrastructure systems. New buildings erected away from existing roads, should meet certain conditions in terms of their accessibility and waste disposal ("Denkschrift über Grundsätze des Städtebaues," 1906, p.5 ff.). Similarly, one can observe that in the past the development of urban quarters followed the characteristic expansion measures arising resulting from the requirements and extension of road, transport and technical infrastructure systems such as the sewage system ("Denkschrift über Grundsätze des Städtebaues," 1906).

In many European cities, including Vienna, vast infrastructural expansion took place during the Wilhelminian period – the so called "Gründerzeit" – particularly with regard to underground town planning. The network of technical infrastructure, especially sewage,

lighting, gas and water networks was extensive. The development of public transport systems, most notably the tram, advanced rapidly during this period too. Urban structures were influenced considerably by the routing of urban supply and waste disposal networks. During the "Gründerzeit", urban design principles adhered to a hierarchical progression from the centre to the periphery of the city. The centre was most well-equipped and enjoyed the greatest benefit of technical infrastructure. With increasing distance from the centre, the provision of supply and waste disposal systems in the outlying quarters became less extensive. Later, as new suburban districts began to be built, these were equipped with the respective technical infrastructure as part of the building measures. Only later were the older suburbs finally connected to the newer technical infrastructure.
The attraction of an urban quarter as a residential district depended on the degree and variety of technical infrastructure available (Behrens, 1971; Weber, 1909). The supply of urban areas with water, sewage systems and energy influences the provisions and attractiveness of a quarter considerably and in some cases may even create the necessary conditions for residential use in the first place. In addition, an urban quarter's connection to public transport networks determines its accessibility and with it the possibilities of interchanges between different locations. The choice of location can therefore be considered as a result of the analysis and evaluation of various criteria (such as situation and availability of facilities).
Applied to a simulation model the local conditions that characterise the urban development correspond to endogenous and exogenous control parameters. By validating the following simulation model using the historical development of the city of Vienna as a basis, it is possible to derive initial conclusions concerning the driving forces and the abstract configuration of a society from the settings of the parameters.

2. From urbanism to suburbanisation

In the development of a city, one can observe phases of growth and shrinkage. A pattern emerges with growth phases following shrinking phases and vice versa. The simplified model of cyclic phases of urban development assumes that the city has a life cycle (Dangschat, 2007). With the help of such models, it is possible to show the centrifugal processes of suburbanism and desurbanism that set in after the initial centralising processes of urbanism. The different urban development cycles and their interaction are shown schematically in the model of cyclic phases of urban development by Van den Berg et al. (1982) (see Fig. 1).

2.1 Urbanism

The European city of the 19th century is a centralised system where the city centre has the best accessibility, the highest prestige value and the most expensive land prices. The social and monetary capital is concentrated in the city centre – furthermore the city centre represented the overall social, political and economic balance of power. The city is experiencing a phase of urbanism (see Fig. 1) – advantages resulting from the agglomeration of the city are used in many different ways, e.g. to increase the concentration of workers and demand, which in turn stimulates an intensive flow of migration from the rural areas to the cities. However, the cities are not prepared for this huge influx of population – as a consequence, public transport systems were lacking, the working hours long, the incomes low, the quickly built housing units small and poorly equipped. The cities had to extend

public transport systems, supply and waste disposal facilities and social infrastructure within a short time. With the development and construction of rail-based public transport systems, the axis-bound growth along these new routes expanded rapidly, followed by the widening of the urban development radius. However, the city centre remained the location with the best accessibility (Fassmann, 2005, p. 33 and p.104; Maier & Tödtling, 2002, pp. 162 ff).

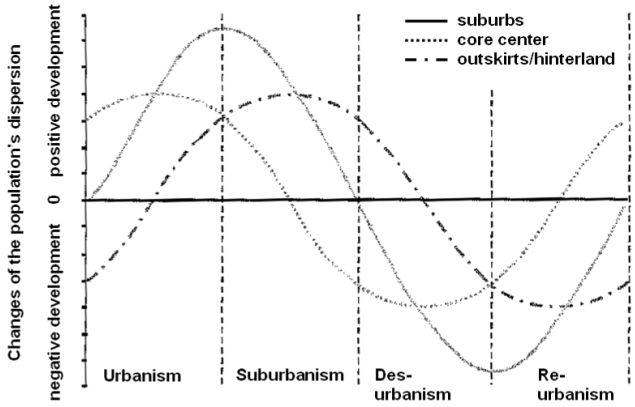


Fig. 1. Model of the cyclic phases of urban development [from Fassmann (2005, p. 105), source: Van den Berg et al. (1982)]

2.2 Suburbanisation

The settlement of the areas between the axes only began after private transport, cars, became widely affordable and available. This lead to a rapid increase in the area occupied by the city and a decrease in the accessibility of the inner city due to a lack of parking spaces and restricted traffic areas. The expansion of the public transport systems, the rising incomes and the high densities in the city centre also contributed to the growth of the city, especially on the outskirts. Thus the population density in the city centre began to decline. In contrast, functions that thrive on agglomeration such as special trade, offices and services settled in the city centre, increasingly displacing residential and industrial uses. A cyclic phase of suburbanisation therefore follows the cyclic phase of urbanism (see Fig. 1).

Suburbanisation can be understood as the relocation of land use and population out of the heart of the city, rural areas or other metropolitan areas into the urban hinterland (Friedrichs, 1981). With the expansion of road networks and the widespread availability of cars as a means of mass transportation for the population it became possible to settle on the outskirts of the city and surrounding areas and still have access to the advantages of the city. The city centre can be reached within a reasonable amount of time, so that settlement in areas with more living space and lower prices became increasingly attractive. The urban realm continues to grow, particularly in the suburbs and at the expense of the city centre. Companies that are not dependent on the agglomeration advantages of the city centre relocate their offices to the outskirts, where more space is available at lower prices. Moreover, as a consequence of suburbanisation sufficient manpower and demand is available in the suburbs. Increasing mobility, newly-built privately-owned homes and the expansion of route networks such as public transport, communication and infrastructure lead to centrifugal, decentralised development (Läpple, 2003)

2.3 De-suburbanisation and Re-urbanism

As the city spreads, the travel distances increase and with it the traffic load and the demand for greater traffic capacity. Due to the relatively low population density in the suburbs, infrastructure-intensive public transport systems are uneconomical, a further reason why an increasing proportion of traffic is dominated by private means of transport. As private means of transport expand, the accompanying pollution has a negative effect on the quality of living in many places, so that more and more inhabitants of the city move to the outskirts. The costs of public transport facilities burden the budget of the city, but because more and more of the affluent population has migrated to the outlying regions, the towns find themselves increasingly in a financial bottleneck (Fassmann, 2005, p. 104; Maier & Tödtling, 2002, p. 163). After the cyclic phase of suburbanisation, a new cyclic trend of urbanism sets in, so-called de-suburbanisation and re-urbanism (see Fig. 1). During de-suburbanisation, cities in the surroundings experience a cyclic phase of urbanism while the respective outlying regions stagnate, followed by a cyclic phase of re-urbanism as activity in the city centres increases.

3. Centripetal and centrifugal forces

The hidden forces described above can be loosely summarised into two opposing forces – a centripetal and a centrifugal force (Krugmann, 1996). This chapter describes how they have been derived. Centripetal forces describe centralising forces, which express the advantages and needs of a dense population. Centrifugal forces describe the advantages of decentralised locations, which are primarily the generous availability of space and the absence of polluting emissions.

Myrdal's (1957) centre-periphery model considers, on the one hand, the emergence of centres and peripheries as a result of deprivation or suction effects, so-called "backwash effects" or centripetal forces. These include effects that arise from the attraction of a dynamically-developing city centre, such as the migration of population or production factors from the rural areas. These effects occur primarily during the cyclic phase of urbanism or re-urbanism (see sections 2.1 and 2.3). On the other hand, the centre/periphery structure is a product of so-called "spread effects". They are also described by the centrifugal forces and express the effects of an expanding centre, such as can be observed in the cyclic phases of suburbanisation or de-suburbanisation (see sections 2.2 and 2.3).

Centripetal and centrifugal forces lead to concentrated or dispersed settlement patterns. The key determining factors are the interdependencies between the location decisions of companies, households and the public authorities. The spatial distribution of activities at a given point in time affect the location conditions for new activities.

The appearance of centripetal forces can be described by means of agglomeration effects. Weber (1909) understands an agglomerative factor as the advantage that results out of the existence of a certain density of a certain land use at a location. Agglomeration effects influence individual economic profits (profit/benefits), but are controlled by other economic entities. They can be divided into localisation effects (Localisation Economies) and urbanisation effects (Urbanisation Economies), although in the following we cover just the urbanisation effects.

Urbanisation effects arise between different actors and between different activities and represent those benefits arising from the entire scope of activities of a region, such as a

widely differentiated employment market, a broad spectrum of public and private services, high-quality communications infrastructure, innovation climate and good connections with and accessibility to other cities (cp. Maier & Tödtling, 2002, p. 101). One of the major urbanisation advantages is the availability of technical infrastructure (Kramar, 2005).
However, in the selection of a location, actors are influenced not only by agglomeration effects but also by opposing deglomeration effects. They describe the negative effects accompanied with overly high density agglomerated areas. As the density increases so too does the cost-benefit ratio. The costs include for example an overloaded city centre with corresponding traffic problems such as traffic jams, noise pollution or high land prices. Agglomeration disadvantages set in from the point at which the costs begin to outweigh the benefits (see Fig. 2).

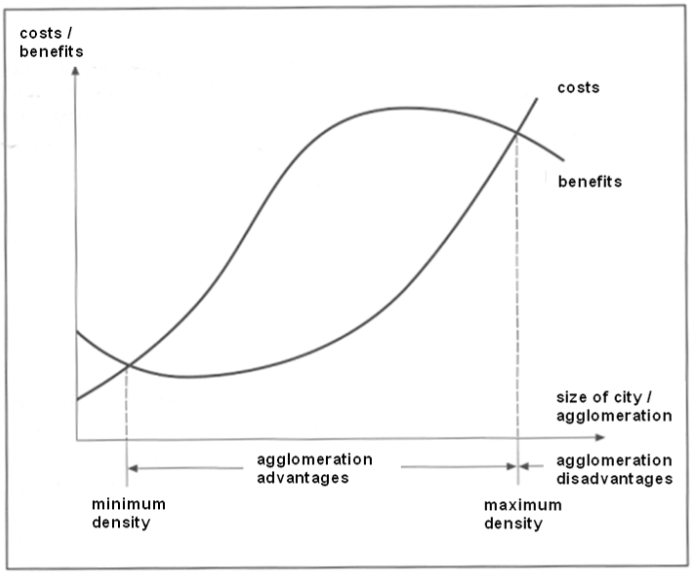


Fig. 2. Agglomeration advantages and agglomeration disadvantages as a function of costs/benefits. Figure from Kulke (2006, p. 242), source: Richardson (1976).

Based on the theoretical concept of the simulation model presented below, the costs curve corresponds to the agglomeration disadvantages. With increasing density the costs increase exponentially. The benefits curve represents the agglomeration advantages. The benefits continue to rise during ongoing aggregation until a saturation point is reached. As agglomeration continues, the usability of a location worsens, relative to its maximum advantageousness (see Fig. 2).
The difference between the benefit curve and the cost curve describes the probability of moving into or away from a location in the simulation model, as illustrated in Fig. 3. If the benefits outweigh the costs, the population density at the location will increase. On the other hand, if the costs outweigh the benefits, the population density at the location will be likely to decrease. During the development of a city (see chapter 2) the apex of the curve (the difference of costs and benefits) moves horizontally (see Fig. 3).

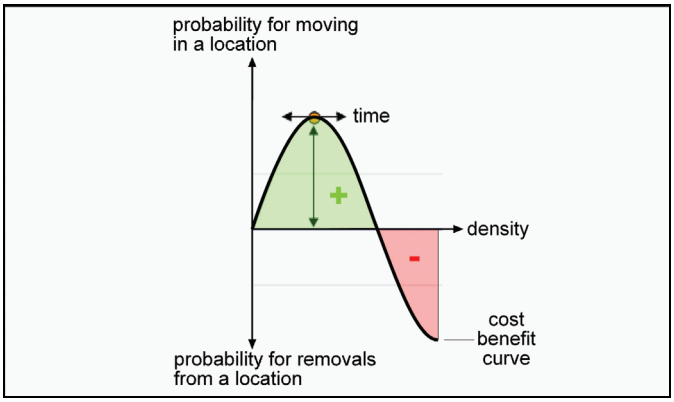


Fig. 3. Probability curve of moving into a location (green) or moving away from a location (red) as a function of density.

4. Modeling and simulation

4.1 Simulation concept

In the design of the simulation model, a key aim was to represent a complex dynamic system through a model that is as simple as possible. The model is intended to be used to explain the emergence of new qualities, to research the complex behaviour of an urban system and to predict future development trends.

Most of the common simulation models describe the growth of urban clusters as aggregation processes by means of DLA (diffusion limited aggregation) and DBM (dielectrical breakdown model) (Batty, 1991), which are both based on the principles of Cellular Automata (CA). In these models, parts (usually residential areas) are successively added to an existing structure by using a probability function. With these simple models, however, it is not possible to consider the change of the compactness of urban clusters over the course of time. For example, urban clusters exhibit rather compact structures during the cyclic phase of urbanism, that shift to become a more ramified structure in the transition from the cyclic phase of urbanism to the cyclic phase of suburbanisation. Furthermore, the specific phenomena for urban agglomerations, such as the restructuring of urban growth, the emergence of new growth zones or the coexistence of urban clusters can only be described by special model extensions (Schweitzer & Schimansky-Geier, 1994). A potential model expansion of the DBM principle exists for example in the formulation of several cell states and the corresponding transformation rules for the simulation of land-use patterns (White & Engelen, 1993). Schweitzer and Schimansky-Geier (1994) have demonstrated with their analysis of the maximum distance between a randomly selected point within the cluster and the next or the most distanced free space, that the shape of the urban residential area is not only controlled by the minimum distance to the city centre but by the minimum distance to the residential border as well. Simple DLA or DBM can be supplemented with such development rules (Schweitzer & Steinbrink, 2002).

Cities are complex systems and follow the principles of self-organization in their development. This hypothesis is a condition for the computer-based simulation model presented here. With the help of this model the occurrence of different phenomena in the

development of residential areas on an urban scale will be reviewed. The underlying analysis of the historical development of the city of Vienna from 1888 to 2001 (Müller, 2005) includes the general hypothesis that improving the accessibility of public transport systems and the supply through technical infrastructure systems leads to an increase in population density. This hypothesis will be verified using the model presented here. Within the model, the population density depends also on the individual weighting of a residential location in relation to the existing population density. This weighting will be based on the probability curve as shown in Fig. 3.

Fig. 4 provides an overview of how the presented hypotheses can be derived from the theoretical basis (see chapter 2 and 3) and how they can be verified using the simulation model described here.

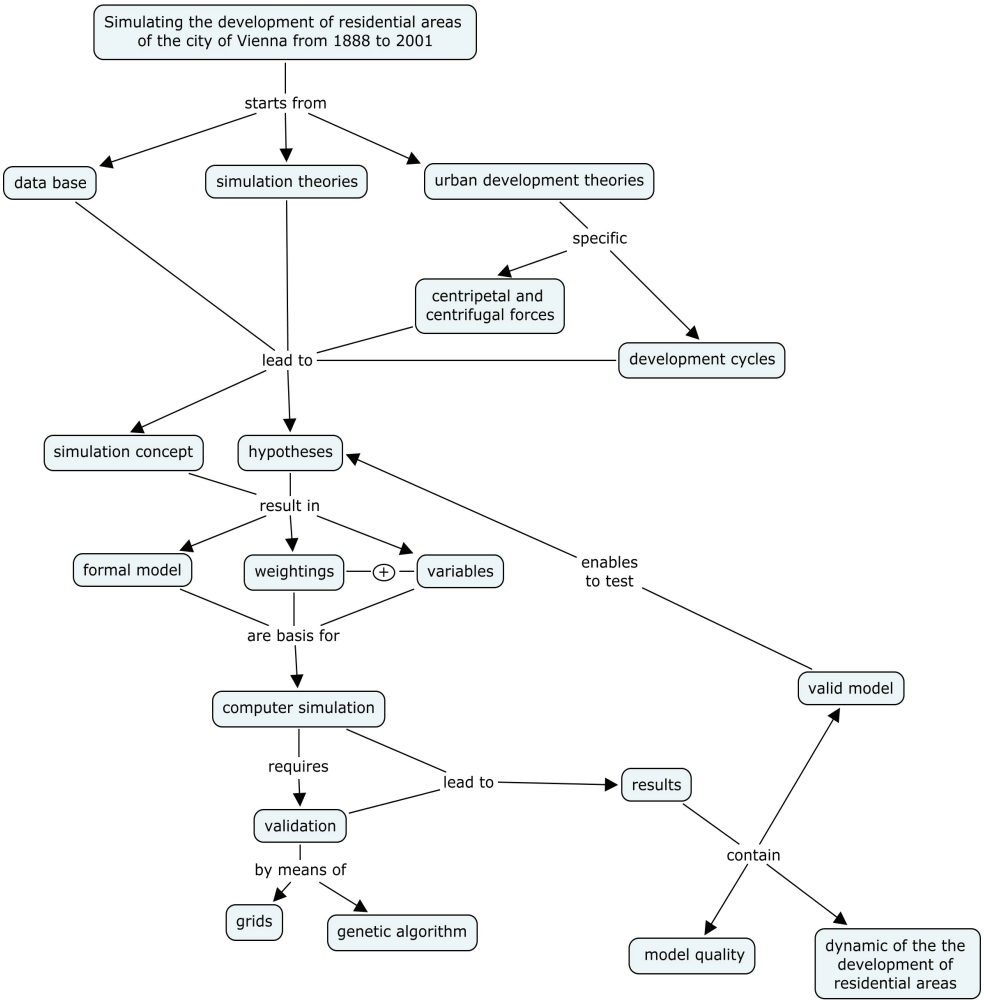


Fig. 4. Overview of simulation concept.

4.2 Data basis and variables

The data basis for the modelling is the analysis of the historical development of the city of Vienna from 1888 to 2001 based on information available from five points in time (1888, 1918, 1945, 1971, 2001). The indicators population density, accessibility of public transport systems and distance to technical infrastructure systems have been reviewed and evaluated.

- *Population density as a measure of the intensity of residential use:* Böventer (1979, p. 17) considers private households as seekers of residences on the one hand and as users of facilities, technical infrastructure and public transport systems on the other. The population density is defined as the sum of inhabitants per quarter hectare.
- *Accessibility of public transport systems as a parameter for exchange and interaction opportunities:* the accessibility of public transport systems is measured as the number of public transport stations within a maximum distance of 750 metres per grid-cell. The location of stations has been calculated using an approximation method, whereby the accessibility of public transport systems has not been determined as a distance dependent function but rather according to distance limits. Consequently the public transport stations are summarised as the public transport opportunities within a certain distance range or zone (Meise & Volwahsen, 1980, p. 129).
- *Supply with technical infrastructure as a measure of local facilities of residential areas:* Technical infrastructure systems include among other things water, sewage, gas, electricity and district heating. The supply with technical infrastructure systems is defined as the density of infrastructural facilities within a radius of max. 1000 metres per grid. Each part of technical infrastructure route that intersects an urban grid cell is a potential infrastructural opportunity for the grid cell under consideration and the surrounding grid cells within a distance of 1000 metres. Thus, for each grid cell of the city of Vienna one can say: The more parts of technical infrastructure routes available in a radius of 1000 metre, the sooner the grid cell is supplied by the appropriate technical infrastructure system. By reaching the maximum value the grid cell under consideration is regarded as completely supplied. The lower the supply density with infrastructural facilities, the greater the potential for future expansion of the technical infrastructure system to achieve full supply.

The data of the three indicators were collected by dividing the city of Vienna into an area of 649 × 519 grid cells of 50 metre edge lengths.

4.3 Simulation

In the simulation model the cell grid of the CA represents the spatial structure of the city. The resolution of the basic grid corresponds to the configuration that was used when ascertaining the data (see section 4.2). Fig. 5 shows the principle for the simulation of the settlement spreading, which is based on the interaction of a so-called potential field and the development (settlement) of an individual area (Batty, 2005, pp. 105-150). In the following, a potential field is used for the calculation of the population density at a location (a grid cell) taking into consideration the population density in the neighbouring locations (neighbouring cells). Which areas of the potential gradient will be settled with a certain probability in the next time step, can be defined using the potential field principle.

For our investigation two potential fields are initiated. The first one represents the population density and the second one is a combination of the supply with infrastructural equipment and public transport accessibility. At each of the five points in time (1888, 1918,

1945, 1971, 2001) the potential of a cell is derived from the processed data on population density, technical infrastructure and public transport accessibility. By linear interpolation between these points in time, the growth rates of the population change as well as the extension of the infrastructural facilities per time step are specified. The location choices of settlement-agents take place endogenously and are represented in the following by a probability function.

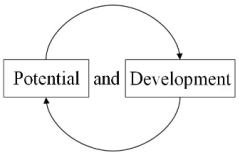


Fig. 5. Interaction of potential and development.

The cells of the model landscape can either be empty or populated with a certain density. With each time step – one month in the present simulation – a certain number of settlement-agents, depending on the growth rate, decide based on their current location preference, which locations they will settle or leave considering the potential values. Afterwards, taking into account the last settlement-activity, the potential values of all cells for the next time step are recalculated.

4.4 Weightings

For the simulation model we use various weightings for several indicators. Based on the statistical analysis of Müller (2005), the weightings for accessibility of public transport systems or for the supply with technical infrastructure are gathered from the results of individual regression analyses. These regression analyses explain at an examined point in time the population density at a specific rate through the accessibility of public transportation and availability of technical infrastructure. The standardised regression coefficients are used as weightings of the particular points in time in the simulation (Table 1 and Table 2). These weightings represent the percentages used to derive the supply potential field from the accessibility of public transportation and technical infrastructure (see equation 1).

	accessibility of public transport facilities	technical infrastructure
1888	50%	50%
1918	51%	49%
1945	26%	74%
1971	48%	52%
2001	58%	42%

Table 1. Weightings of the accessibility of public transportation and technical infrastructure.

The factor technical infrastructure is made up from the variables water, sewage, gas, and electricity. In a factor analysis it can be proven that these variables can be explained by the factor technical infrastructure (Müller 2005). The specific indicators of technical infrastructure (water, sewage, gas, and electricity) from the factor loadings out of the factor

analysis are available for the weightings at the current point in time and can be used in the simulation model (see equation (1)).

	gas	water	canal	electricity	district heating
1888	24%	37%	39%	0%	-
1918	25%	24%	27%	24%	-
1945	26%	23%	26%	24%	-
1971	25%	25%	25%	25%	18%
2001	25%	25%	25%	25%	15%

Table 2. Weightings of the indicators for technical infrastructure.

4.5 Formal model

For the formal representation of the model we agree on the following conventions. A cell of the CA is indicated with $H = \{1, 2, \ldots, H_N\}$.The domain of the variable population density per cell D^H is normalised, i.e. scaled on the interval between 0 and 1 (D^H = [0, 1]). In the same way the domains of the supply variables per cell are normalised and indicated as follows: District heating F^H = [0, 1], gas G^H = [0, 1], sewage K^H = [0, 1], water W^H = [0, 1], electricity S^H = [0, 1], public transportation system O^H = [0, 1]. From a summary of the supply variables the values of the supply potential field (V^H = [0, 1]) result:

$$V^H(t+1)=\omega_T \cdot \left(\omega_F \cdot F^H(t)+\omega_G \cdot G^H(t)+\omega_K \cdot K^H(t)+\omega_W \cdot W^H(t)+\omega_S \cdot S^H(t)\right)+\omega_O \cdot O^H(t). \quad (1)$$

The factor ω indicates the fraction of a supply variable on the supply potential field V^H. The values of a supply variable for a cell H at time step t are calculated by linear interpolation of the collected data at the specified points in time and are taken as exogenous influence variables for the model. The variable for the population potential field is indicated with P^H = [0, 1]. The potential values are calculated with the following equation:

$$P^H(t+1) = \frac{1}{5} \cdot \left(\left(\sum_{B \subset U(H)} P^B(t) \right) + D^H(t) \right). \quad (2)$$

The index B indicates a cell of the subset $U(H)$, which consists of the four directly neighbouring cells of a considered cell H without the considered cell H itself. The potential field is recalculated after each time step. The rates for positive R_p and negative R_n population growth results from the sums of the differences of the population densities of the individual cells between two points of time:

$$r^H = M^H(T+1) - M^H(T). \quad (3)$$

The points of time of the data assessment are declared with T = {1888, 1918, 1945, 1971, 2001}. The time steps between these points in time are indicated with t. In our case a time step covers one month. In contrast to the normalised population density D, the absolute population of a cell is indicated with M^H. The absolute change of a cell's population density between the points in time of data collection is declared with r^H. The growth rates R at a time step t depend on the scaling of model time, which is defined by the parameter c:

$$R_n(t) = \left(\sum_H r^H \mid r^H, r^H < 0 \right) \Big/ c \cdot \left((T+1) - T \right). \tag{4}$$

$$R_p(t) = \left(\sum_H r^H \mid r^H, r^H > 0 \right) \Big/ c \cdot \left((T+1) - T \right). \tag{5}$$

The differences of the population densities r^H are added to the negative growth rates (R_n), if $r^H < 0$, and otherwise, if $r^H > 0$, r^H is added to the positive growth rates (R_p). If the model time is scaled to month, for the denominator in equation 4 and 5 the value $12 * (1918 - 1888) = 360$ results for the first simulation period.

At which cells the population density rises or falls is defined by assessment curves for moving away from or moving into individual inhabitants. These curves indicate which areas are evaluated as the least attractive ones by the current inhabitants due to the density and supply values of these locations and therefore that a decline in population is probable, or which areas are evaluated as the most attractive ones by the current inhabitants for the same reasons and therefore that an increase of population is probable. The probability ρ for moving away from a cell or moving to a cell of an inhabitant results from the assessment functions ρ_D for the population density and ρ_V for the supply potential of a cell:

$$\rho^H(t) = \left(\rho_D^H(t) + \rho_V^H(t) \right) \Big/ 2. \tag{8}$$

Following Krugmann (1996), the assessment function for the population density (ρ_D) results from a trade-off between two exponential functions for calculation of a centripetal F_{petal} and a centrifugal F_{fugal} force:

$$\rho_D^H(t) = F_{petal}^H(t) - F_{fugal}^H(t). \tag{7}$$

To simplify equation 7 we use the beta distribution[1] for the calculation of ρ_D which allows one to model an assessment curve with 2 instead of 4 control parameters. This will be relevant in chapter 6 where the search for optimal parameter settings is described. Since we assume different assessment curves for moving away from a cell or moving to a cell, equation 6 is to be distinguished in two cases, which in the following are indicated with ρ_{D_out} for the moving-away-from-curve and with ρ_{D_in} for the moving-to-curve:

$$\left.\begin{aligned} \rho_{D\_out}^H(t) &= \frac{1}{B(p_{out}, q_{out})} \cdot x^{p_{out}-1}(1-x)^{q_{out}-1} \\ \rho_{D\_in}^H(t) &= \frac{1}{B(p_{in}, q_{in})} \cdot x^{p_{in}-1}(1-x)^{q_{in}-1} \end{aligned}\right\}. \tag{8}$$

[1] cf. Wikipedia: http://en.wikipedia.org/wiki/Beta_distribution (last visited at 09.07.2008)

The course of the assessment curve[2] is defined by means of the parameter p_{out} and q_{out} as well as p_{in} and q_{in}. The term $B_{(p,\ b)}$ indicates the beta function respectively for moving away from a cell and for moving to a cell:

$$B(p,q) = \int_0^1 x^{p-1}(1-x)^{q-1}\,dx. \tag{9}$$

We will deal with the concrete significance of the assessment curves in the following chapter 5.3. The assessment function for the supply-potential is a positive linear function to the supply-potential of a cell:

$$\rho_V^H(t) = V^H(t). \tag{10}$$

A location is regarded as being more attractive, the better its supply is. Based on the probability ρ, which is calculated for each cell, the decisions of the inhabitants where they move away from and where they move to can be made with the help of the roulette wheel method (Goldberg, 1989, S. 231). To choose a value ρ from the H_N values ρ^1, ρ^2, ρ^3… ρ^{HN}, the size of each probability value is specified by its weighting (size of the slot). At roulette wheel selection this weighting is indicated as selection probability w^H and can be calculated by dividing each value by the sum of all values:

$$w^H = \rho^H \Big/ \left(\rho^1 + \rho^2 + \rho^3 + ... + \rho^{H_N}\right) \quad oder \quad w^H = \rho^H \Big/ \sum_H \rho^H . \tag{11}$$

To choose a value H the following algorithm is executed:

a. generate a random value z between 0 and 1
b. set $sum = 0$
c. for $H = 1$ to H_N do
   begin
      $sum = sum + w^H$
      if ($sum \geq z$) Then return H
   end.

The randomly chosen value z corresponds to the position of the roulette ball and step c) tests in which slot H it ends up.

4.6 Computer program

The formal model presented in the previous chapter 5.2 was implemented in the programming language Delphi as a stand-alone simulation program for windows. Its basic functions are shown in Fig. 6.

The main area of the program window (Fig. 6, A) contains the graphic output, the visual representation of the various values of the separate variables which are saved per cell such as, for example, the population and supply density as well as the rates of moving away from or moving to. The darker the cells are (Fig. 6, A), the higher the density values. The green and red marked cells represent locations that inhabitants move to or move away from. The menu bar on the right-hand side of the program window is arranged in three register cards.

[2] An interactive animation of the beta distribution is available at:
http://www.uni-konstanz.de/FuF/wiwi/heiler/os/vt-beta.html (last visited at 09.07.2008)

Fig. 6, B shows the weightings for equation 1 and Fig. 6, C the parameters for the beta distribution of equation 8 for the definition of the course of the assessment curve for moving away from and moving to. The diagram shows the corresponding assessment curves for moving away from (red curve) and moving to (green curve).

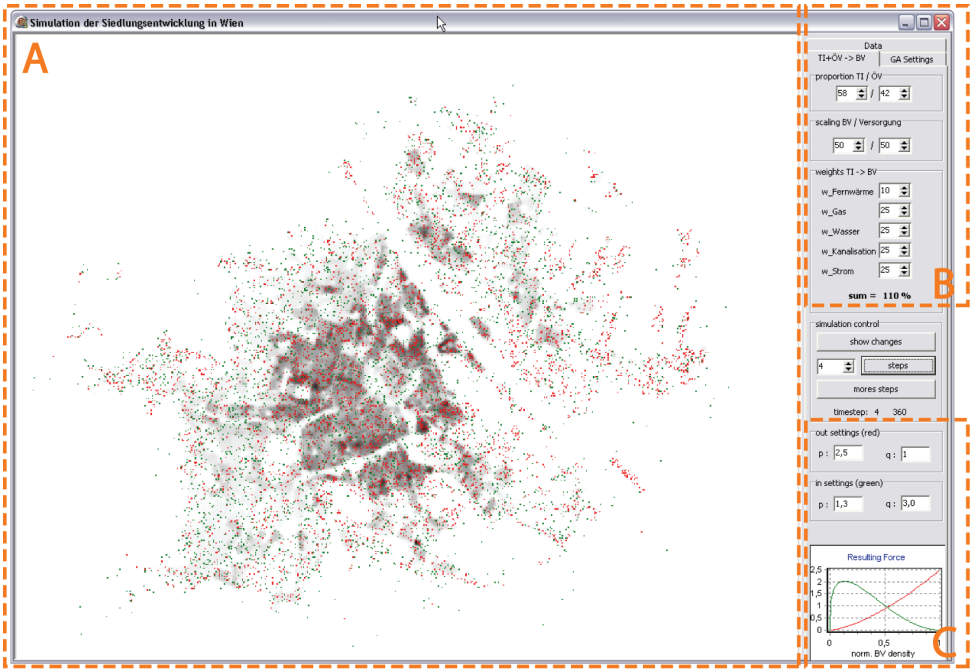


Fig. 6. Screenshot of the simulation program. Area A shows the population density in the year 2001. The considered period for moving away from or moving to in 5 time steps.

The calculation of the rates R_n and R_p for the population growth (see equations 4 and 5) are shown in Fig. 7. The first row (delta BV) declares the absolute population growth in the

Changes

	1888-1918	1918-1945	1945-1971	1971-2001
delta BV	648957	-288376	-74636	-69578
total rate/t	1803	-890	-239	-193
+ BV	1095641	725661	584335	322985
+ rate/t	3043	2240	1873	897
- BV	-446571	-1014279	-658781	-392638
- rate/t	-1240	-3131	-2111	-1091
steps	360	324	312	360
+ big_BV	1095641	725661	584335	322985
+ big_rate/t	3043	2240	1873	897
- big_BV	-446571	-1014279	-658781	-392638
- big_rate/t	-1240	-3131	-2111	-1091

Fig. 7. Screenshot of the rates of change.

periods of the corresponding columns. The second row (total rate/t) comprises the interpolated absolute monthly growth rate per time step t. The third row (+BV) shows the absolute number of influxes in a period. In the fourth row (+rate/t) the influxes per time step $R_p(t)$ are declared. The fifth row (-BV) shows the absolute number of movements away from locations in a period. In the sixth row (-rate/t) the movements away from locations per time step $R_n(t)$ are declared. In the seventh row (steps) the number of time steps t (in months) of the respective period are declared.

5. Validation

The validity of a simulation run is judged at the end of a period by means of the population distribution. One cannot expect the simulated data and surveyed data of population density to correspond at a detailed level (e.g. per cell) because of the abstraction for example of topographic, social and economic conditions. Consequently, the correspondence between the population densities is measured in radial areas, which are defined by the distance to the city centre (centre of mass). The position of a cell H in a coordinate system can be defined as vector v^H. The centre of mass Z can be calculated by forming the sum of the weighted vectors and dividing it by the total number of settled cells (Schweitzer & Steinbrink, 2002). The weight of a vector corresponds to the population density of a considered cell:

$$\rho_V^H(t) = V^H(t)\,. \tag{12}$$

The population density is measured in concentric circles around the centre of mass Z (Fig. 8, left). Now the population differences between real data and the results of the simulations inside the measure rings are calculated. The population difference summed up for all rings represents the quality of the model. The model quality is best if the absolute sum of the population difference is as low as possible. Based on the real data a simulation run starts at T-1 and ends at T. Therefore the model quality refers always to the period under consideration.

Fig. 8 (right-hand side) shows the various measure charts. Both charts above show the measurements of fractal dimensions. Diagram A represents the differences of separate measure cells. Diagram B shows the measure rings, which are used for the validation of the simulation. The blue graph represents the differences per ring and the red graph shows the sum of the differences.

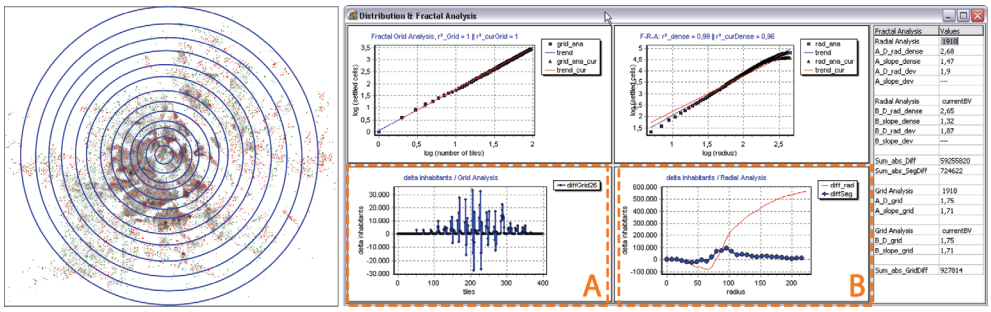


Fig. 8. Validation method.

The result of the simulation respectively the quality of the model that results after a simulation run from T-1 to T depends mainly on the settings of the control parameters p and q for the definition of the assessment curve by means of equation 8. The values for the control parameters are kept constant during a simulation period (from T-1 to T). The challenge for the validation of the present model was to search for the optimal settings for these control parameters for each considered period. Our aim was to achieve the best possible model quality and therefore validity for each period.

5.1 Genetic algorithm

To find the optimal settings for the control parameters a genetic algorithm was used. The base forms the structure of the so-called simple genetic algorithm (SGA) according to Goldberg (1989, p. 69), which was adapted for the present challenges. The initial population for the SGA are 32 individuals that consist of a randomly generated chromosome of 28 binary numbers. In these chromosomes the four control parameters for the assessment curves of equation 8 are coded (Fig. 9). For each individual, a simulation period is run with the corresponding control parameters and the corresponding model quality is calculated. This quality represents the evaluation criteria for the suitability of a control parameter setting and can therefore be used for the fitness definition of an individual.

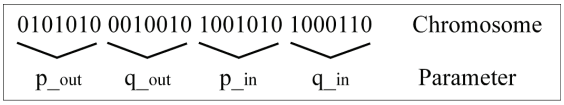


Fig. 9. Coding of the parameters in a chromosome.

After the simulation period has been run through for each of the 32 individuals and a fitness value has been assigned to each of them, a new generation of individuals is generated. The probability for the reproduction of an individual is defined by its fitness value. The fitness values are scaled in such a way that the individual with the highest value is selected with twice the probability as an individual with an average fitness. The individual with the lowest fitness has no chance of reproduction.

The crossing rate indicates the probability that the selected individuals are crossed or copied without a change into the new generation. Crossing means the mixture or recombination of two (parent) chromosomes (Fig. 10). At the present SGA we use a crossing rate of 80%.

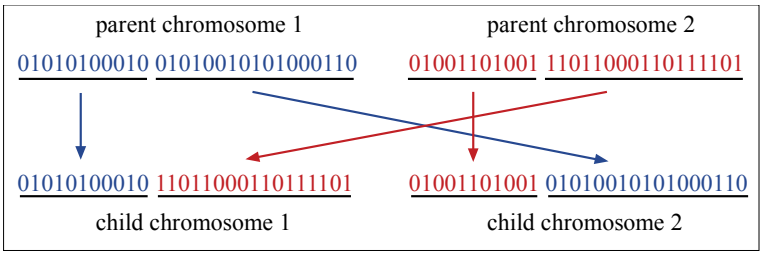


Fig. 10. Generation of child chromosomes by crossover of the parent chromosomes.

The mutation rate indicates the probability for the inversion of a randomly selected binary number of a chromosome, e.g. from 0 to 1. At the present model we have set the mutation rate to 1%. Therefore, on average, 1 out of every 100 binary numbers mutates.

5.2 Grids

With the available data we have to consider all possible locations (cells) of the cell landscape for moving to a cell or moving away from a cell of inhabitants per time step *t*. The cell landscape includes 649 × 519 = 336,831 cells. For the first period from 1888-1918 per time step *t*, 3,043 influxes and 1,240 effluxes have to be calculated. This results in 4,283 calculations per *t*, which in turn means 1,542,212 calculations for the whole period. With current computers, the simulation of this period lasts approximately 4.5 hours. For one simulation per period this would be an acceptable time. It becomes problematic, however, when in order to search for the optimal control parameter settings many hundreds of periods have to be calculated (see section 6.1), since the overall computing time can quickly add up to several weeks. For this reason we vary the resolution of the cell landscape denoted by the grid in the following. To change the resolution, the existing population data of several 50 × 50 metre cells are merged in a larger cell. The original cell raster with an edge length of 50 m per cell is denoted as grid 1. At grid 5, 5 × 5 original cells are merged to a cell with a side length of 250 m and at grid 10, 10 × 10 original cells are merged to a cell with a side length of 500 m and so on (Fig. 11).

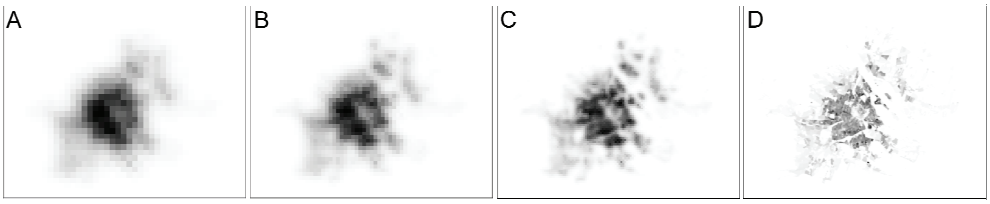


Fig. 11. Representation of the population density with various grid resolutions. A) grid 15, B) grid 10, C) grid 5, D) grid 1.

The use of various grids allows a comparison of simulation results at different degrees of data precision. Does it make any difference whether we have population density data for grid 20 or grid 1 and do the resulting optimal control parameters differ only slightly? This question is answered in the description of the results in the next chapter.

In the search for optimal control parameters, as a first step, the optimal values for the control parameters for the coarsest grid (grid 20) are calculated. In a second step the results of these calculations are integrated in the initial population for the next finer grid (grid 15) to calculate the optimal control parameters for this grid. These steps are repeated for grid 10, grid 5, and grid 1. We act on the hypothesis that the transfer of the optimal values from the coarser to the finer grids decrease the necessary number of generations and with this the overall computing time necessary for the search by means of a genetic algorithm. Whereas up to 100 generations were calculated at grid 20, the search for the optimal values was aborted after 5 generations per period at grid 1. Nevertheless at grid 1 a model quality between 75% and 90% could be achieved for all periods (Fig. 12).

6. Results and conclusions

6.1 Model quality

In chapter 6 the model quality was defined as totalised absolute difference between simulated and real population density over the individual measuring rings. The best model quality (100%) is achieved when simulated and real population densities match exactly. The model qualities achieved for the various periods with regard to the different grids are shown in Fig. 12.

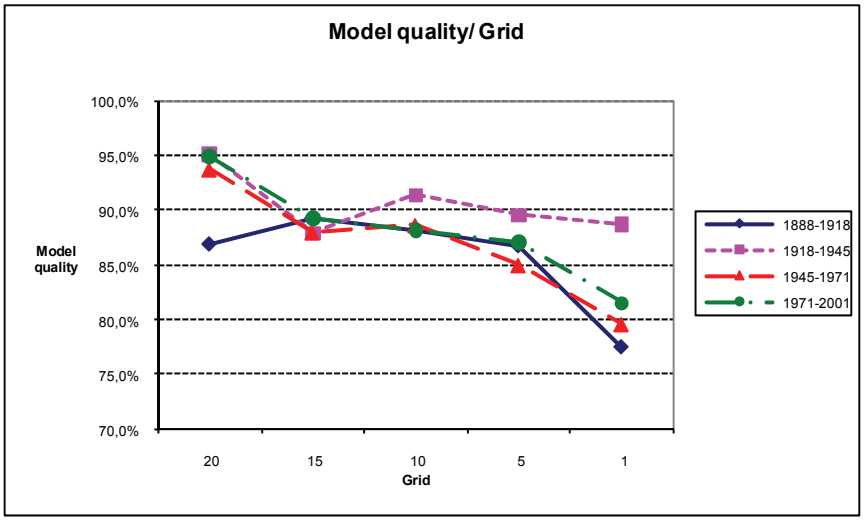


Fig. 12. Model quality (ordinate) at optimal values for the four control parameters for the various periods depending on different grids (abscissa).

In principle for all periods the achieved model quality is very good with values over 85% for grids greater than 1. There are several possible reasons for the quite obvious decrease in quality for all periods for grid 1: firstly, the measuring method used and its accuracy were adapted for each grid. Secondly, the small number of calculated generations for grid 1, since the optimal parameters do not agree with those of the coarser grids (see Fig. 13) and therefore could probably be optimised by calculating further generations. Though, this was not possible within this project for reasons of time. Thirdly, the limiting of the maximum value (upper bound) for the four control parameters at 12.7 has possibly led to some distortions, since the calculated optimal values for q_{in} at T_3 and T_4 have already reached this maximum (see Fig. 13).

6.2 The valid model

The most important results of the project are the optimal values for the four control parameters p_{out} and q_{out} as well as p_{in} and q_{in} that are found by means of the genetic algorithm. The various coloured bars in Fig. 13 show the values of the control parameters, which have led to the best results at the different grids (best model quality).

In Fig. 13 it is clearly visible that at each period some parameter values change only slightly while others change very strongly if we compare the different grids. For example the values for q_{out} and p_{in} at T_1 in Fig. 13, character A are relatively similar from grid 20 to grid 1, whereas in the same period the values for p_{out} vary quite significantly from grid 20 to grid 1. The variations of individual parameters lead to relevant differences in the course of the assessment curves for moving away from or moving to. These differences in the course of the assessment curves become apparent if we compare, for example, the mean values of the control parameters for the four periods with the optimal values for grid 1, as shown in Fig. 14 to Figure 16.

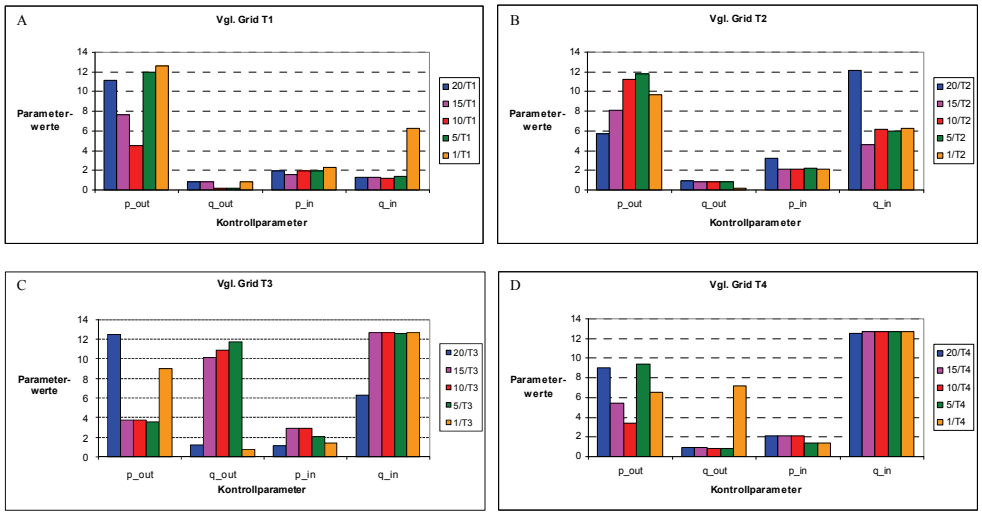


Fig. 13. Comparison of the optimal control parameters per time step T at different grids (grid 20 to grid 1 in the legend). A) T1=1888-1918, B) T2=1918-1945, C) T3=1945-1971, D) T4=1971-2001

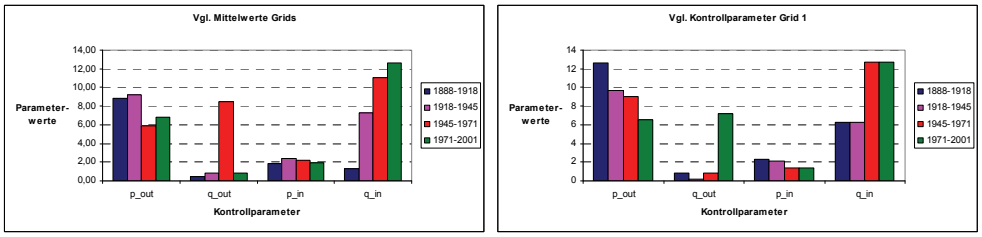


Fig. 14. Development of the control parameters. Left: Mean values of the optimal values of the four control parameters for different grids per period T (cf. Fig. 13). Right: Optimal control parameters for grid 1.

The diagram on the left in Fig. 14 shows the mean values of the optimal values for the four control parameters p_{out}, q_{out}, p_{in} und q_{in} for different grids per period T1 to T4 with strongly divergent values for q_{out} in the third period and for q_{in} in the course of all considered periods (T1 to T4). At the optimal parameter values for grid 1 in Fig. 14 right, the values for q_{in} vary very little from each other in both first and in both last periods. On the other hand, during the same period the values for p_{out} vary considerably from each other. For q_{out} the strongest variation is in the period from 1971 to 2001.

Using the mean values (Fig. 14, left), the assessment curves for moving to and moving away from are created, which are shown in Fig. 15. From an interpretation of these assessment curves we can conclude that, for example, at T_3 in Fig. 15, character C, the population has moved from the outer urban districts to rural areas, whereas at T_4 in Fig. 15, character D, the population redistribution was from the city centre to rural areas.

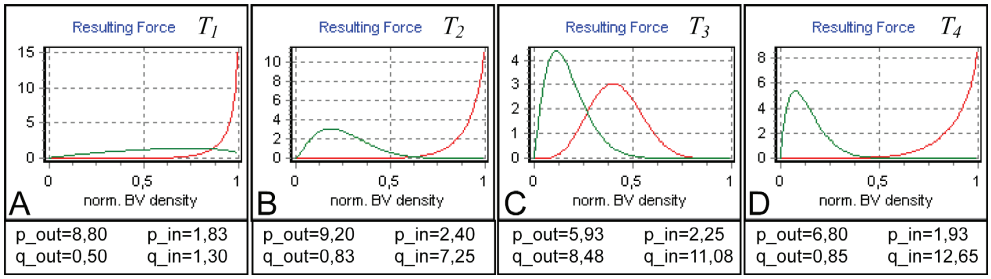


Fig. 15. Assessment curve for the mean values from Fig. 14, left. A) T1=1888-1918, B) T2=1918-1945, C) T3=1945-1971, D) T4=1971-2001

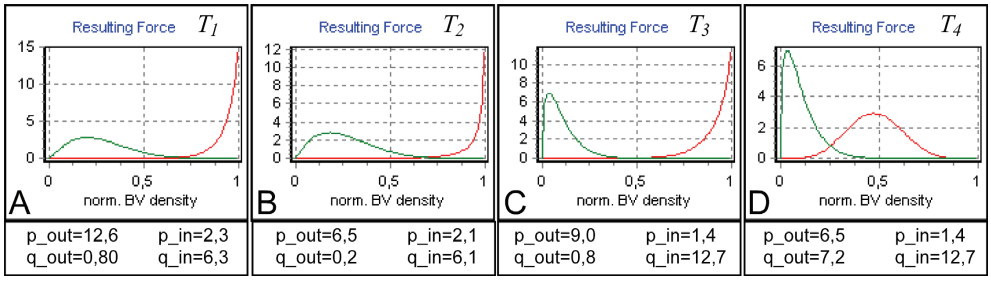


Fig. 16. Assessment curve for the best values for grid 1 from Figure 13. A) T_1=1888-1918, B) T_2=1918-1945, C) T_3=1945-1971, D) T_4=1971-2001

The assessment curves for grid 1 in Fig. 16, character C and character D, have to be interpreted just contrarily. The curves show that the population moves from outer urban areas to rural areas at T_4. However, a comparison of the green moving to curves of the diagrams in Fig. 15 and Fig. 16 we can observe that they both show a similar course. The interpretation of the moving to curves is essentially the same in Fig. 15 and Fig. 16.

If we consider the differences described between the graphs in Fig. 15 and Fig. 16 we can conclude on the one hand that the exactness of the data (in the present application represented by the cell size) has a considerable effect on the results of the simulation. Consequently, the significance of the results of the present study are relative, since they are related to the population and supply data from a 50 × 50 m grid (see chapter 4.2)
However, if we look at the similarities between the assessment curves in Fig. 15 and Fig. 16 we can conclude on the other hand that even with a much coarser grid and correspondingly less accurate data, the values for the control parameters can be calculated with sufficient exactness and acceptable significance.

6.3 The dynamics of settlement development

Apart from the development of a valid model we could have shown the suitability of the four control parameters p_{out} and q_{out} as well as p_{in} and q_{in} (see equation 8) as formal representations of the individual assessment of a residential location concerning the existing population density. With the simulation, the hypotheses (see chapter 4.1) could be supported whereby an improvement in the accessibility of public transport facilities and the proximity to technical infrastructural facilities causes an increase in population density. The assessment curves resulting from the control parameters of grid 1 allow a visual interpretation of the calculated values for these parameters.
Considering the green curves in the diagrams in Fig. 16 it can be observed that the preferred locations for influxes have shift more and more from the outskirts of the town (at T_1 and T_2) to the rural areas (at T_3 and T_4). The population settles increasingly in those areas with lower population density. As a result the suburbanisation process (see chapter 2.2) in Vienna from 1888 to 2001 can be identified in the shift of the green curves maximum in the diagrams in Fig. 16. Suburbanisation is most clearly visible between the Second World War and the last considered point in time. This process is also apparent in the consideration of the red curves in Fig. 16. There we see that the movements away concentrates first on the city centre (from T_1 to T_3), shifting only in the last period to the outer urban districts (at T_4). The overall illustration of moving to and moving away from (Fig. 17) over the course of time confirms our interpretation of the assessment curves.
At the first point in time T_1 the migration from the exurban fringe already becomes apparent. The population is moving to mainly sparsely settled areas – at T_1 primarily beyond the former town's fortification. Over the course of time – until the last point in time T_4 – the population moves increasingly away from the centrally-located densely-populated urban areas. The progressive suburbanisation process becomes more and more obvious since the population increasingly prefers sparsely populated or empty areas as residential areas. We can observe an overall spread of the settlement towards as well as beyond the city limits.
In chapter 3 we have already indicated that the position of the maximum and the form of an assessment curve for movements to and away depends on various economic, social, cultural and ecological general conditions. For the present simulation model the form of the curves was determined only for four periods (Fig. 16). The assumption that the assessment curves dynamically change over the course of time is more realistic. According to this, starting at *T-1,* a curve of a diagram in Fig. 16 would gradually turn into the same coloured curve at

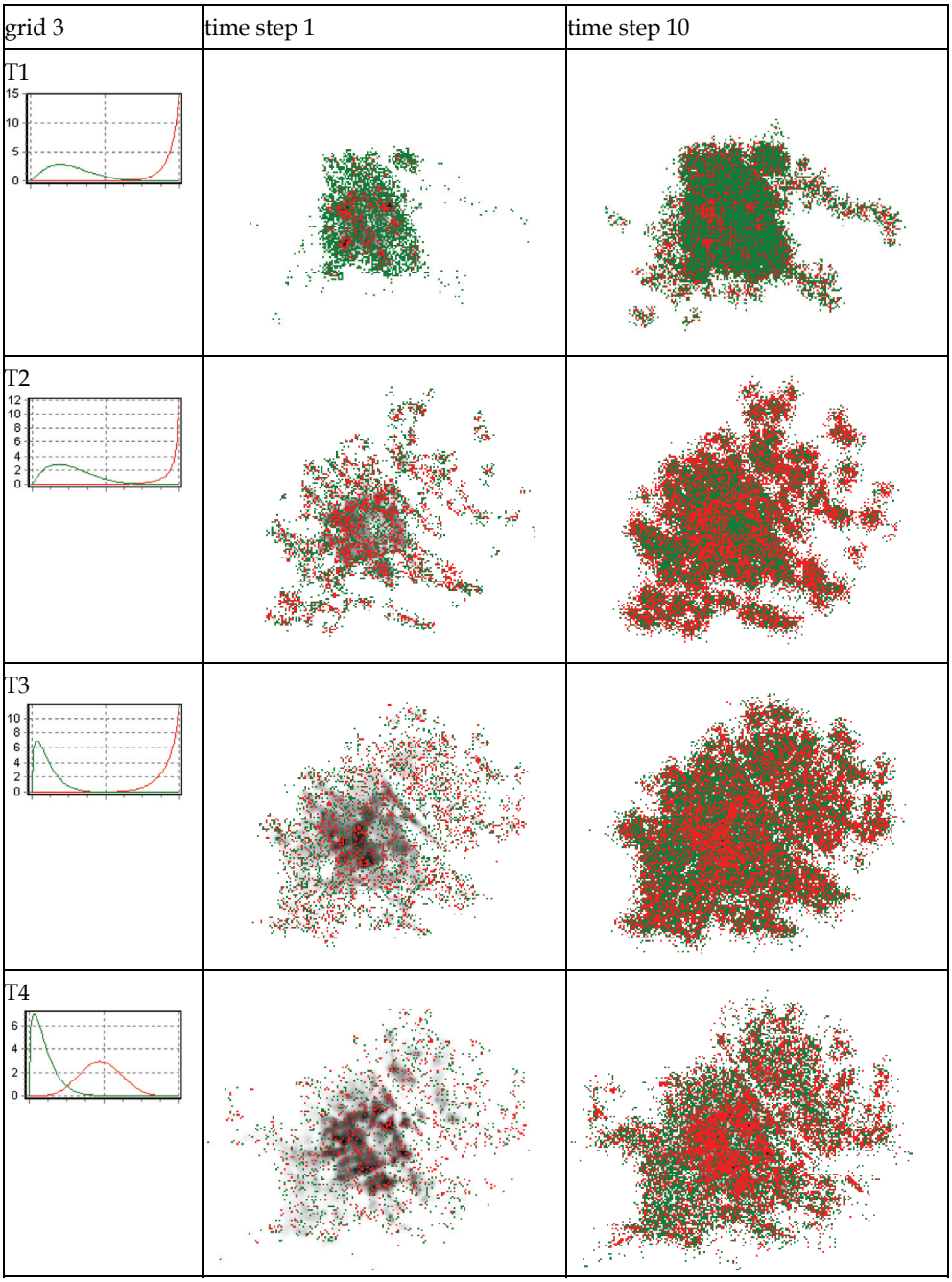


Fig. 17. Overall illustration of the movements to and movements away after one and ten time steps (second and third column) for different periods (rows T1 to T4)

point *T* in time. The concept for the change of the assessment curves over the course of time provides a robust explanatory approach for describing the basic processes which are responsible for the change of settlement structures.

7. Summary and outlook

The relatively high model quality indicates that we have achieved a high validity with the model shown here for simulating the population distribution of the city of Vienna from 1888 to 2001. With the help of four control parameters for the definition of the assessment curves we have developed a sensible and robust concept for the explanation of the driving forces of urban development processes. Through the exogenous definition of the infrastructural supply of a location we could show that the development of the population density can be essentially regulated by infrastructure investments. Furthermore the suburbanisation processes, which have taken place over the course of the considered period in the city of Vienna, can be explained by our model.

The simulation model was designed to be as simple and transparent as possible to enable a sensitivity analysis of the control parameters and through this to ensure the comprehensibility of the results. Only through the construction of comprehensible models will it be possible to improve the acceptability of simulations for political decision makers or urban and regional planners. Similarly, only models whose operating principles are described transparently can be improved further and further in the course of its usage. The next steps for the development of the model are to first explain the extension of the technical infrastructure and the spread of the public transport facilities on the basis of the development of the population density, and secondly to adapt the model in a way that the development of the population density can be circularly coupled with the availability of technical infrastructure and the spread of public transport facilities. By adding supplemental exogenous parameters such as, for example, topographical conditions, the significance of the model could be increased still further.

Furthermore, one could examine to what extent changes to the control parameter values in the present simulation can be connected with characteristic values of the economic and technological development of the city of Vienna during the investigated period.

The application perspective for the model at hand is to estimate the consequences of particular planning activities on the basis of restricted scenario models. Such a scenario could, for example, include planned infrastructure expansion or the effects of increasing transportation costs on the assessment curves and with this on the development of the settlement structure.

8. Acknowledgements

This research project was financed by the *Jubiläumsstiftung* of the city of Vienna.

9. Referneces

Batty, M. (1991). Generating urban forms from diffusive growth. *Environment and Planning A, 23*(4), 511-544.

Batty, M. (2005). *Cities and Complexity: Understanding Cities with Cellular Automata, Agent-Based Models, and Fractals*. London: MIT Press.

Behrens, K. C. (1971). *Allgemeine Standortbestimmungslehre*. Opladen: Westdeutscher Verlag GmbH.

Böventer, E. (1979). *Standortentscheidung und Raumstruktur*. Hannover: Schroedel Verlag.

Dangschat, J. (2007). Reurbanisierung – eine Renaissance der (Innen-)Städte? *Städtepolitik und Stadtentwicklung, 3*.

Denkschrift über Grundsätze des Städtebaues. (1906). In R. Baumeister (Ed.), *Beiträge zum Städtebau*: Association of German Architects and Engineers.

Fassmann. (2005). *Stadtgeographie I*. Braunschweig: Westermann

Friedrichs, J. (1981). *Stadtanalyse: Soziale und räumliche Organisation der Gesellschaft*. Opladen: Westdeutscher Verlag.

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning* (1 ed.). Boston: Addison-Wesley.

Kramar, H. (2005). Innovation durch Agglomeration: Zu den Standortfaktoren der Wissensproduktion. In D. Bökemann (Ed.), *Wiener Beiträge zur Regionalwissenschaft* (Vol. 20). Wien.

Krugmann, P. (1996). *The Self-Organizing Economy*. Cambridge, Mass.: Blackwell.

Kulke, E. (2006). *Wirtschaftsgeographie*. Paderborn: Schöningh Verlag

Läpple, D. (2003). Thesen zu einer Renaissance der Stadt in der Wissensgesellschaft. In N. Gestring, H. Glasauer, C. Hannemann, W. Petrowsky & J. Pohlan (Eds.), *Jahrbuch Stadt/Region* (pp. 61–78). Opladen: Leske und Budrich.

Lichtenberger, E. (1986). *Stadtgeographie: Begriffe, Konzepte, Modelle, Prozesse* (Vol. 1). Stuttgart: Teubner.

Maier, G., & Tödtling, F. (2002). *Regional- und Stadtökonomie: Standorttheorie und Raumstruktur*. Wien, New York: Springer Verlag.

Meise, J., & Volwahsen, A. (1980). *Stadt- und Regionalplanung, Ein Methodenhandbuch*. Braunschweig: Friedr. Vieweg & Sohn Verlagsgesellschaft

Müller, D. (2005). *Wien 1888 – 2001: Zusammenhänge der Entwicklung der technischen Infrastruktur- und ÖV-Systeme in den Siedlungsgebieten*. Wien: Peter Lang Verlag.

Myrdal, G. (1957). *Economic Theory and Underdeveloped Regions*. London: Duckworth.

Richardson, H. W. (1976). Growth Pole Spillovers: the dynamics of backwash and spread. *Regional Studies. The Journal of the Regional Studies Association, 10*(1), 1-9.

Schweitzer, F., & Schimansky-Geier, L. (1994). Clustering of Active Walkers in a Two-Component System. *Physica A, 206*, 359-379.

Schweitzer, F., & Steinbrink, J. (2002). Analysis and Computer Simulation of Urban Cluster Distribution. In K. Humpert, K. Brenner & S. Becker (Eds.), *Fundamental Principles of Urban Growth* (pp. 142-157). Wuppertal: Müller + Busmann.

Van den Berg, L., Drewett, R., Klaassen, L. H., Rossi, A., & Vijverberg, C. H. T. (1982). *Urban Europe: A Study of Growth and Decline* (Vol. 1). Oxford: Pergamon Press.

Weber, A. (1909). *Über den Standort der Industrie*. Tübingen: J.C.B. Mohr Verlag

White, R., & Engelen, G. (1993). Cellular automata and fractal urban form: a cellular modelling approach to the evolution of urban land-use patterns. *Environment and Planning A, 25*(8), 1175-1199.

3

Spatial Dynamic Modelling of Deforestation in the Amazon

Arimatéa C. Ximenes, Cláudia M. Almeida, Silvana Amaral,
Maria Isabel S. Escada and Ana Paula D. Aguiar
National Institute for Space Research,
General Coordination for Earth Observation,
Av. dos Astronautas, 1758, PO Box 515, São José dos Campos, SP
Brazil

1. Introduction

New GIS technologies have been employed to support public policies and actions towards environmental conservation, aiming to preserve biodiversity and mitigate the undesirable side-effects of human activities. The spatio-temporal simulation of systems dynamics is an example of such new technologies and helps scientists and decision-makers to understand the driving forces lying behind processes of change in environmental systems. In assessing how systems evolve, it is possible to figure out different scenarios, given by diverse socio-economical, political and environmental conditions (Soares-Filho et al., 2001), and hence, anticipate the occurrence of certain events, like land cover and land use change, including deforestation. According to Openshaw (2000), computer simulation models provide qualitative and quantitative information on complex natural phenomena. In this sense, spatial dynamic models may be defined as mathematical representations of real-world processes or phenomena, in which the state of a given place on the Earth surface changes in response to changes in its driving forces (Burrough, 1998).

Spatial dynamic models are commonly founded on the paradigm of cellular automata (CA). Wolfram (1983) defines CA as "[…] mathematical idealisations of physical systems in which space and time are discrete, and physical quantities take on a finite set of discrete values. A cellular automaton consists of a regular uniform lattice (or 'array'), usually infinite in extent, with a discrete variable at each site ('cell'). […] A cellular automaton evolves in discrete time steps, with the value of the variable at the site being affected by the values of variables at sites in its 'neighbourhood' on the previous time step. The neighbourhood of a site is typically taken to be the site itself and all immediately adjacent sites. The variables at each site are updated simultaneously ('synchronously'), based on the values of the variables in their neighbourhood at the preceding time step, and according to a definite set of 'local rules'." (Wolfram, 1983, p. 603).

This work applies a CA model – Dinamica EGO – to simulate deforestation processes in a region called *São Félix do Xingu*, located in east-central Amazon. EGO consists in an environment that embodies neighbourhood-based transition algorithms and spatial feedback approaches in a stochastic multi-step simulation framework. Biophysical variables

drove the simulation model of the present work, and statistical validation tests were then conducted for the generated simulations (from 1997 to 2000), by means of multiple resolution fitting methods. This modelling experiment demonstrated the suitability of the adopted model to simulate processes of forest conversion, unravelling the relationships between site attributes and deforestation in the area under analysis.

2. Study area

The region of *São Félix do Xingu* is regarded as one of the current three main occupation fronts in the Brazilian Amazon. Recent official data indicate that *São Félix do Xingu* was the Amazonian municipality owning the highest deforestation rates at the end of last decade and the beginning of this decade. According to Becker (2005), *São Félix do Xingu* and other occupation fronts represent the new inland Amazonian frontiers, namely mobile frontiers, which differ from the frontiers observed in the early stages of human occupation in this region in the 1970s with regard to three aspects: i) the prevailing migration is intra-regional, and mostly rural-urban; ii) the private capital plays a crucial role in such fronts, which present a great diversity of local actors, mainly loggers, cattle raisers, and grains producers, and iii) these fronts own greater accessibility and connectivity, relying on a denser (air, terrestrial, and fluvial) transportation network as compared to the one available in the 1970s. Recent data on deforestation show that *São Félix do Xingu* was also the Amazonian municipality presenting the greatest absolute values of deforested areas between the years 2000 and 2006. Out of the total deforested area assessed in the Brazilian Legal Amazon in the years 2005 (665,854 km²) and 2006 (679,899km²), *São Félix do Xingu* alone accounted for 13,626 km² (2.0%) and 14,496 km² (2.1%), respectively (Brazilian National Institute for Space Research or *Instituto Nacional de Pesquisas Espaciais* [INPE], 2006). Part of the occupation history in this region can be ascribed to natural rubber and mahogany exploitation, mining, cattle raising, and huge private and public rural settlements projects, among which cattle raising is the chief economic activity (Escada et al., 2007).

The study area comprises most of the *São Félix do Xingu* municipality and its surroundings, located in the State of *Pará* (PA), east-central Amazon, north of Brazil. The *Xingu* river, one of the major tributaries of the Amazon river, crosses *Pará* in the longitudinal direction (Fig. 1). The data used in this research are set in the Polyconic Projection System, Datum SAD-69, between longitude coordinates 52° 30′ and 51° 00′W, and latitude coordinates 05° 52′ and 07° 07′S. Besides the municipality of *São Félix do Xingu*, the study area also comprises the municipal seats of *Ourilândia do Norte* and *Tucumã*, and part of the municipalities of *Marabá*, *Parauapebas*, and *Água Azul do Norte*. Within *São Félix do Xingu*, the study area contains the following villages and districts: *Vila Taboca*, *Nereu*, *Tancredo Neves*, *Carapanã*, *Minerasul*, and *Ladeira Vermelha*. Part of the Indian reserves *Apyterewa*, *Kayapó,* and *Xinkrin do Cateté* are also included in the study area.

The portion of *São Félix do Xingu* embraced by the study area presents already a consolidated occupation. In the 1980s, this area sheltered pioneer fronts, marked by intense land occupation processes concentrated along the PA-279 road, which connects *Tucumã* to *São Félix do Xingu* (Shimink & Wood, 1992; Research Network for Environmental Modelling in the Amazon [GEOMA], 2004). This region is currently characterised by the presence of large farms, landed property concentration, and very often, illegal practices of land ownership (Escada et al., 2005).

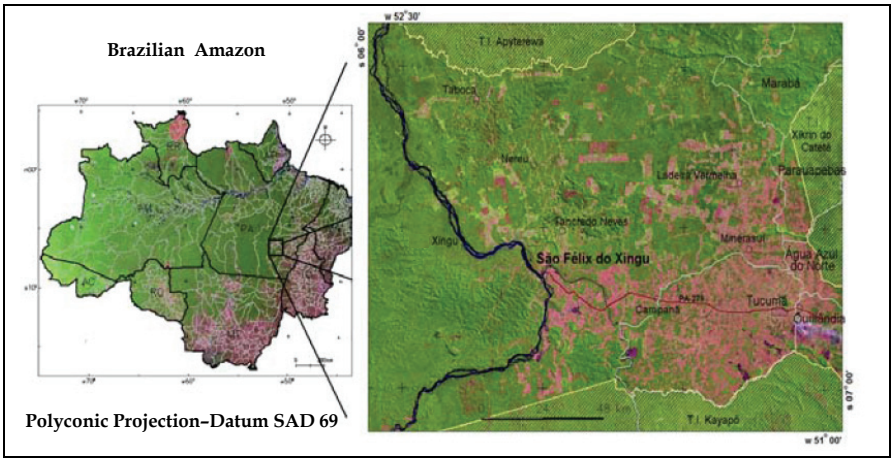


Fig. 1. Location of study area: *São Félix do Xingu* municipality and surroundings

Although this region has cattle raising as the leading economic activity, ore exploitation (mainly cassiterite and gold) has had an important role in its economy and regional spatial structure since the 1970s. Initially, ore was transported along the *Xingu* river and its tributaries, and also through small aircrafts (Santana, 2007). From the first half of the 1990s onwards, mining activities entered into decline, leaving behind them a dense network of roads connecting farms in the region, which considerably reduced the importance of the air and fluvial transport (Amaral et al., 2006; Escada et al., 2007).

3. Input data for the spatial dynamic model of deforestation

A platform for the spatio-temporal modelling of landscape dynamics – Dinamica EGO – was used to generate the simulations of deforestation in the analysed study area. Dinamica EGO consists in a cellular automata environment that embodies transition algorithms operating commonly through a Moore neighbourhood (3 x 3 window) as well as spatial feedback approaches in a stochastic multi-step simulation framework. Dinamica EGO is a free domain platform and was developed by the Centre for Remote Sensing of the Federal University of Minas Gerais – CSR-UFMG[1] (Soares-Filho et al., 2002, 2009; Rodrigues et al., 2007).

Real data on deforestation from 1997 to 2000 drove the simulation model, helping to assess the total amount of forest conversion into other land cover classes in the study area for the given period of analysis. A set of explaining variables related to deforestation together with internal parameters of Dinamica EGO were jointly combined to generate a simulation for the year 2000. In the following sections, methodological procedures employed at each stage of the modelling process (data acquisition, variables selection, exploratory analysis, calculation of transition probabilities, parameterisation, and accuracy assessment) will be dealt with in a thorough manner.

[1]The CA modelling platform Dinamica EGO is available for download at http://www.csr.ufmg.br/dinamica/

3.1 Deforestation data

The original deforestation map, containing the land cover classes forest, grasslands, rivers, deforested areas until 1997, and deforested areas from 1997 to 2000, was acquired from the digital PRODES (Brazilian Deforestation Assessment Project) database (INPE, 2006) and is shown in Fig. 2. This map and other input data were pre-processed in the software IDRISI Kilimanjaro (Eastman, 2003).

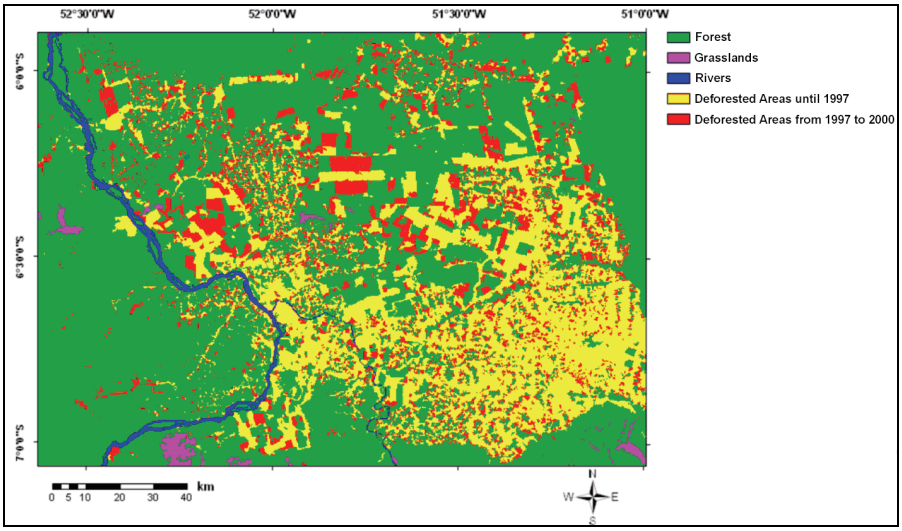


Fig. 2. PRODES deforestation map for the study area from 1997 to 2000

This PRODES deforestation map was reclassified, so as to generate the land cover maps for the initial time of simulation – year 1997 (Fig. 3), and for the final time of simulation – year 2000 (Fig. 4). Considering that PRODES methodology does not take into account deforestation over natural grasslands, this work similarly restricted itself to simulate deforestation only over forested areas. Due to generalisation procedures adopted in the PRODES maps, only the *Xingu* river and its major tributary, *Fresco* river, are visible, since minor streams were disregarded in face of the spatial resolution adopted in the maps (120 x 120 m).

3.2 Explaining variables

The proper choice of a set of explaining variables is critical for the success of a spatial dynamic model. The forested cells suitability to undergo deforestation precisely depends on the relations between such variables and the response variable (deforestation). In this experiment, the analysed variables were selected based upon similar previous studies (Alves, 2002; Laurance et al., 2002; Aguiar et al., 2007; Soares-Filho et al., 2006; Pereira et al., 2007; Brandão Júnior et al., 2007), which report the prevailing variables in deforestation processes. The factors which influence deforestation are manifold. However, the difficulties in data acquisition limit the input data actually employed in the modelling process. Six biophysical variables were evaluated: i) distance to paved roads; ii) distance to non-paved roads; iii) distance to urban centres; iv) distance to rivers; and v) distance to previously deforested areas in 1997.

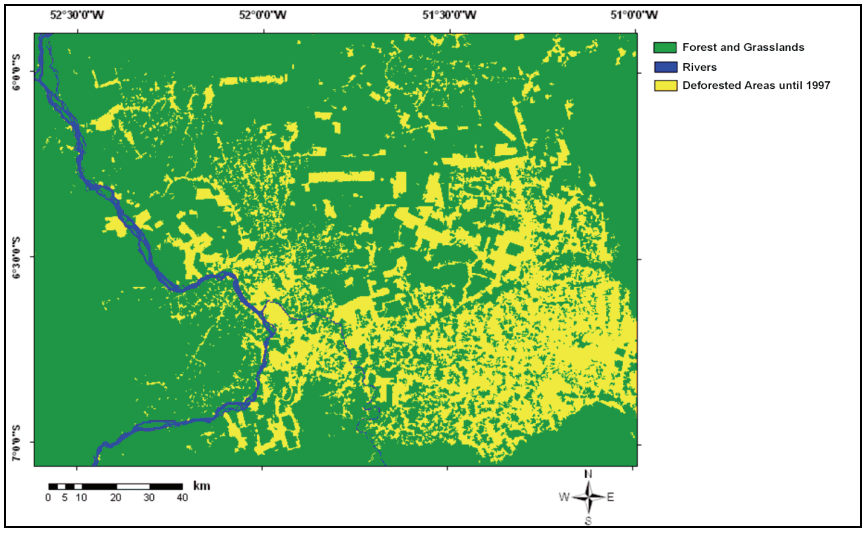


Fig. 3. Land cover map in 1997, derived from the PRODES deforestation map reclassification

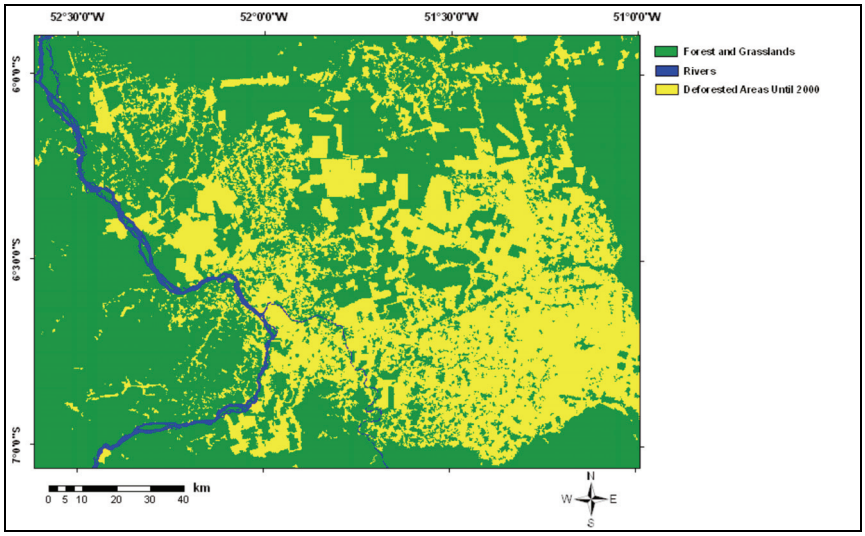


Fig. 4. Land cover map in 2000, derived from the PRODES deforestation map reclassification

By means of a visual analysis of the original deforestation map in 1997, it was possible to observe that both the distance to roads and to urban centres have a limited influence on deforestation processes. In 1997, the occupation in the study area was already well-established, reducing the availability of land for deforestation close to roads and urban settlements. Although the fluvial transport plays a minor role in the region, as previously exposed, it was observed the mushrooming of deforestation patches in the vicinities of the *Xingu* and *Fresco* rivers during the analysed period, which although occurred to a reduced

extent in farther areas, justified the inclusion of the variable 'distance to rivers' in the model. In CA models, two types of variables can be used: i) the static ones, which are kept constant throughout the model run, and ii) the dynamic ones, that suffer changes throughout the successive time steps, which are then continuously updated at each iteration. Both of them were built upon basis of the PRODES reclassified map in the year 1997 (Fig. 3). The static variable corresponds to the map of distance to rivers (Fig. 5), and the map of distance to previously deforested areas in 1997 (Fig. 6) entered the model as a dynamic variable. In order to categorise the grids of distances, i.e. generate optimal discrete ranges of distances, special automatic routines available in the Dinamica EGO were used, which are based on algorithms of lines generalisation (Agterberg & Bonham-Carter., 1990; Goodacre et al., 1993).

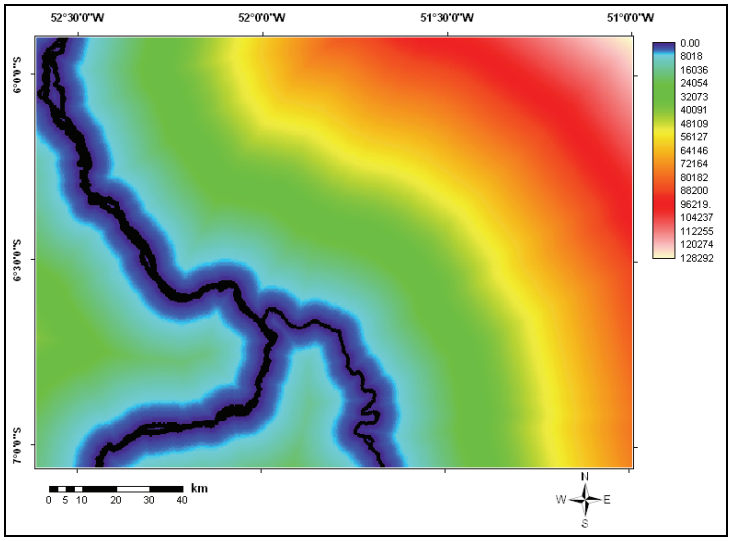


Fig. 5. Colour-sliced map of distance to rivers (static variable), defined in meters

4. The spatial dynamic modelling experiment

4.1 Exploratory analysis

The statistical method 'weights of evidence' was employed for the parameterisation of this modelling experiment. Such method is entirely based on Bayes´s theorem, also known as the conditional probability theorem, which assumes the independence of events. In this sense, the eventual existence of spatial dependence (or spatial association) between pairs of explaining variables must be initially verified. For this end, two statistical indices were used: the Cramer´s Coefficient (V) and the Joint Information Uncertainty (U) (Bonham-Carter, 1994). Both of them are based on the ratio of overlapping areas among the different categories (in this case, ranges of distance) belonging to two maps of explaining variables (or evidences). The Cramer´s Coefficient operates with absolute values of area, while the Joint Information Uncertainty deals with relative (percentage) values, and hence, tends to be more robust than the former index, for it avoids the risk of bias represented by absolute area values.

Bonham-Carter (1994) reports that values less than 0.5 either for V or U suggest less association rather than more. In this way, the threshold of 0.5 was adopted to decide whether a variable should remain in the model (V or U < 0.5) or be excluded from it (V or U ≥ 0.5). In this work, the Cramer's Coefficient and the Join Information Uncertainty presented low values (V = 0.21 and U = 0.0485), indicating that both variables could be simultaneously employed in the model.

4.2 Global transition rates

The global transition rates refer to the total amount of change in the simulation period, regardless of its spatial distribution, i.e. without taking into account spatial peculiarities at the local level, which are those related to biophysical and infrastructural characteristics of each cell in the study area.

In this modelling experiment and in other experiments where the initial and final land cover maps are available, the transition rates were calculated by a cross-tabulation operation between the land cover maps from 1997 and 2000, producing as output a transition matrix with land cover change rates observed during this period.

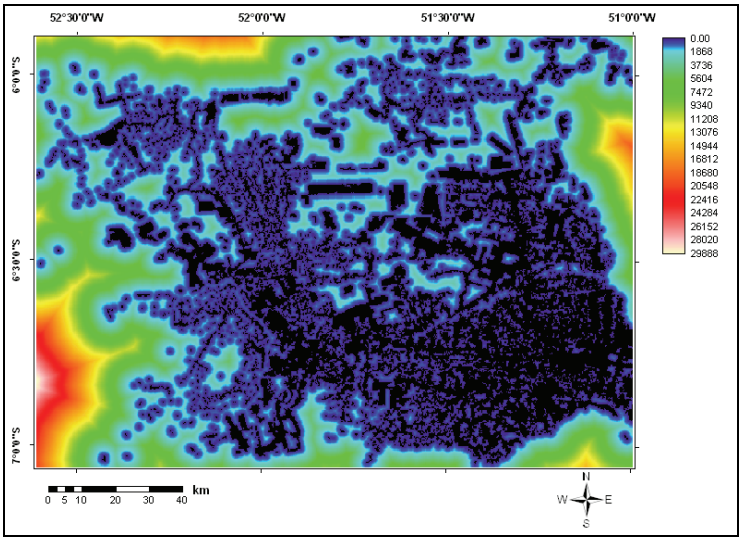


Fig. 6. Colour-sliced map of distance to previously deforested areas in 1997 (dynamic variable), defined in meters

4.3 Local transition probabilities

The local transition probabilities, different from the global transition rates, are calculated for each cell considering the natural and anthropic characteristics of the site. For estimating the land cover transition probabilities in each cell, represented by its coordinates x and y, an equation converting the *logit* formula into a conventional conditional probability was used. The *logit* corresponds to the natural logarithm of odds, which consists in the ratio of the probability of occurring a given land cover transition to its complementary probability, i.e.

the probability of not occurring the transition. This concept derives from the Bayesian weights of evidence method, from which the land cover transition probability can be obtained through algebraic manipulations of the *logit* formula, as follows (Bonham-Carter, 1994):

$$P(T_i^{\alpha}/V_i^1,...,V_i^{m_\alpha})=O(T_i^{\alpha})\cdot e^{\sum_{v=1}^{m_q} W_{1,v}^{+}} / 1+\sum_{\alpha=1}^{\eta} O(T_i^{\alpha})\cdot e^{\sum_{v=1}^{m_q} W_{1,v}^{+}} \quad (1)$$

where P corresponds to the probability of transition in a cell; i corresponds to a notation of cells positioning in the study area, defined in terms of x,y coordinates; α represents a type of land cover transition, e.g. from a class c to a class k, within a total of η transitions (in this particular experiment, there is only one type: from forest to deforested areas); V_i^1 corresponds to the first variable observed in cell i, used to explain transition α; $V_i^{m_\alpha}$ corresponds to the m-th variable observed in cell i, used to explain transition α; O (T_i^{α}) represents the odds of transition T^{α} in the i-th cell, expressed by the ratio of the probability of occurrence of T_i^{α} over its complementary probability, i.e., P (T_i^{α})/ P ($\overline{T}_i^{\alpha}$); and $W^{+}_{i,v}$ corresponds to the positive weight of evidence for the i-th cell regarding the v-th variable range, defined as:

$$W_{i,v}^{+}=\log_e P(V_i^{m_\alpha}/T_i^{\alpha})/P(V_i^{m_\alpha}/\overline{T}_i^{\alpha}) \quad (2)$$

where P ($V_i^{m_\alpha}$/ T_i^{α}) is the probability of occurrence of the m-th variable range observed in cell i, used to explain transition α, in face of the previous presence of transition T_i^{α}, given by the number of cells where both $V_i^{m_\alpha}$ and T_i^{α} are found divided by the total number of cells where T_i^{α} is found; and P($V_i^{m_\alpha}$/ $\overline{T}_i^{\alpha}$) is the probability of occurrence of the m-th variable range observed in cell i, used to explain transition α, in face of the previous absence of transition T_i^{α}, given by the number of cells where both $V_i^{m_\alpha}$ and $\overline{T}_i^{\alpha}$ are found, divided by the total number of cells where T_i^{α} is not found.

The W^{+} values represent the attraction between a determined landscape transition (in this case, from forest to deforested areas) and a certain variable range. The higher the W^{+} value is, the greater is the probability of a certain transition to take place. On the other hand, negative W^{+} values indicate lower probability of a determined transition in the presence of the respective variable range. Using the W^{+} values concerning the several distances ranges of the static and dynamic variables employed in the analysis of deforestation, the Dinamica EGO model calculates the cells transition probabilities according to Equation (1). The grid cells are assigned a value of probability and a probability map is then generated. In order to evaluate if the model is well calibrated, i.e. if the employed explaining variables are appropriate and if the categorisation of the numerical grids is optimal, this map must present the area with the highest transition probability values as close as possible to the areas that actually underwent deforestation processes.

4.4 Defining the dinamica EGO internal parameters

Dinamica EGO incorporates two empirical land cover allocation algorithms (or transition functions) called: expander and patcher. The expander function accounts for the expansion of previous patches of a certain land cover class. The patcher function, on its turn, is designed to generate new independent patches (seed cells), which are not physically

connected with previous patches of the same land cover class (Soares-Filho et al., 2002). In summary, the expander function performs transitions from a state i to a state j only in the adjacent vicinities of cells with state j. And the patcher function performs transitions from a state i to a state j only in the adjacent vicinities of cells with state other than j.
Based on a visual analysis of the PRODES deforestation map (Fig. 2) during the analysed period (1997-2000), it was not observed the occurrence of deforestation by means of diffusion processes, i.e. amidst the virgin forest, what is emulated by the patcher function. In this way, only the expander function was employed in the model, which simulates the formation of deforestation patches through the expansion of previously deforested areas, as exposed above.
The average size and variance of size (in hectares) of the new deforestation patches are also required as internal parameters by Dinamica EGO. The definition of these parameters is done heuristically. The ideal average size (μ) was set to 300 ha, and the variance (σ^2) to 500 ha. The model contains another heuristically determined parameter, which is the so-called 'patch isometry index (PII)'. This index represents a numerical value ranging from 0 to 2, which is multiplied by the probability values of the eight cells belonging to the Moore neighbourhood, before the application of the transition function. A high isometry index results in compact patches, while low values are reflected in more fragmented landscape patterns. In this modelling experiment, an isometry index of 1.5 was adopted, what represents a balance between compactness and fragmentation. This value generates results in compliance with the deforestation pattern observed in the study area, which presents a mixture of geometrically stable (compact) deforestation patches produced by capitalised farmers that use tractors for the forest clear-cut, as well as fragmented patches generated by small farmers, deprived of sophisticated means for the forest removal.

5. Validation

For assessing the accuracy of the CA simulation model performance, fuzzy similarity measures applied within a neighbourhood context were used. Several validation methods operating on a pixel vicinity basis have been proposed (Costanza, 1989; Pontius, 2002; Hagen, 2002, 2003; Hagen-Zanker et al. 2005; Hagen-Zanker, 2006), aimed at depicting the spatial patterns similarity between a simulated and a reference map, so as to relax the strictness of the pixel-by-pixel agreement. The fuzzy similarity method employed in this work is a variation of the fuzzy similarity metrics developed by Hagen (2002), and has been implemented in the DINAMICA model by the CSR team.
Hagen´s method is based on the concept of fuzziness of location, in which the representation of a cell is influenced by the cell itself and, to a lesser extent, by the cells in its neighbourhood. Not considering fuzziness of category, the fuzzy neighbourhood vector can represent the fuzziness of location. In the fuzzy similarity validation method, a crisp vector is associated to each cell in the map. This vector has as many positions as map categories (land cover classes), assuming 1 for a category = i, and 0 for categories other than i. Thus, the fuzzy neighbourhood vector (V_{nbhood}) for each cell is given as:

$$V_{nbhood} = \begin{bmatrix} \mu_{nbhood\ 1} \\ \mu_{nbhood\ 2} \\ \vdots \\ \mu_{nbhood\ C} \end{bmatrix} \quad (3)$$

$$\mu_{nbhood\ i} = | \mu_{nbhood\ i,1} * m_1 , \mu_{crisp\ i,2} * m_2 , \ldots, \mu_{crisp\ i,N} * m_N |_{Max} \quad (4)$$

where $\mu_{nbhood\ i}$ represents the membership for category i within a neighbourhood of N cells (usually $N=n^2$); $\mu_{crisp\ ij}$ is the membership of category i for neighbouring cell j, assuming, as in a crisp vector, 1 for i and 0 for categories other than i ($i \subset C$); m_j is the distance-based membership of neighbouring cell j, where m accounts for a distance decay function, for instance, an exponential decay ($m = 2^{-d/2}$), with *d* being the unitary distance measured in between two cells centroids). The selection of the most appropriate decay function and the size of the window depend on the vagueness of the data and the spatial error tolerance threshold (Hagen, 2003). As it is intended to assess the model spatial fit at different resolutions, besides the exponential decay, a constant function equal to 1 inside the neighbourhood window and to 0 outside can also be applied. Equation (5) sets the category membership for the central cell, assuming the highest contribution is found within a neighbourhood window n x n. Next, a similarity measure for a pair of maps can be obtained through a cell-by-cell fuzzy set intersection between their fuzzy and crisp vectors:

$$S(V_A, V_B) = [\ |\mu_{A,1}, \mu_{B,1}|_{Min}, |\mu_{A,2}, \mu_{B,2}|_{Min}, \ldots, |\mu_{A,i}, \mu_{B,i}|_{Min}\ ]_{Max} \quad (5)$$

where V_A and V_B refer to the fuzzy neighbourhood vectors for maps A and B, and $\mu_{A,i}$ and $\mu_{B,i}$ are their neighbourhood memberships for categories $i \subset C$ in maps A and B, as in Equation (6). According to Hagen (2003), since the similarity measure S (V_A,V_B) tends to overestimate the spatial fit, the two-way similarity is instead applied:

$$S_{twoway}(A, B) = |S(V_{nbhoodA}, V_{crispB}), S(V_{crispA}, V_{nbhoodB})|_{Min} \quad (6)$$

The overall similarity of a pair of maps can be calculated by averaging the two-way similarity values for all map cells. However, when comparing a simulated map to the reference map (real land cover in the final time of simulation), this calculation carries out an inertial similarity between them due to their areas that did not suffer any change. To avoid this problem, the CSR team introduced a modification into the overall two-way similarity method of Dinamica EGO, using two maps of differences, which present value 1 for the cells that underwent change, and 0 for those that did not change. In this way, each type of change is analysed separately using pair-wise comparisons involving maps of differences: (i) between the initial land cover map and a simulated one, and (ii) between the same initial land cover map and the reference one. This modification is able to tackle two matters. First, as it deals with only one type of change at a time, the overall two-way similarity measure can be applied to the entire map, regardless of the different number of cells per category. Second, the inherited similitude between the initial and simulated maps can be eliminated from this comparison by simply ignoring the null cells from the overall count. However, there are two ways of performing this function. One consists of counting only two-way similarity values for non-null cells in the first map of difference, and the other consists in doing the opposite. As a result, three measures of overall similarity are obtained, with the third representing the average of the two ways of counting. As random pattern maps tend to score higher due to chance depending on the manner in which the null cells are counted, it is advisable to pay close attention to the minimum overall similarity value. This method has proven to be the most comprehensive when compared to the aforementioned methods, as it yields fitness measures with the highest contrast for a series of synthetic patterns that depart from a perfect fit to a totally random pattern (Soares-Filho et al., 2009).

6. Results and discussion

Table 1 presents the transition matrix resulting from the cross-tabulation operation between the initial (1997) and the final (2000) land cover maps. It provides the percentages of forest conversion, what corresponds in the particular case of this work to the global transition rates from forest to deforested areas, calculated as 13.8%. The classes 'deforested areas' and 'rivers' did not suffer any change during the study period.

	Land Cover Classes	2000		
		Forest/ Grasslands	Deforested Areas	Rivers
1997	Forest/Grasslands	0.861641846	**0.138358154**	0
	Deforested Areas	0	1	0
	Rivers	0	0	1

Table 1. Land cover transition matrix for the period from 1997 to 2000

According to what was previously explained in Section 4.3, the calculation of the local transition probabilities, i.e. the probabilities of land cover change for each cell, is based on the values of the positive weights of evidence (W^+). Tables 2 and 3 present the values of W^+ for each distance range of the dynamic variable 'distance to previously deforested areas in 1997' and to the static variable 'distance to rivers', respectively. Fig. 7a and 7b graphically present the behaviour of the W^+ values in relation to the successive distance ranges of these two explaining variables.

The curve of W^+ for the variable 'distance to previously deforested areas in 1997' (Fig. 7a) reveals the concentration of weights with the greatest values in the first distance ranges, what demonstrates the predominance of new deforestation patches in the surroundings of pioneer areas (Alves, 2002; Aguiar et al., 2007). Said in other words, the pattern of deforestation expansion in the study area during the analysed period mostly presents patches of large extensions, following a trend to take place in the vicinities of previously deforested areas, also in face of the reported landed property concentration in this region.

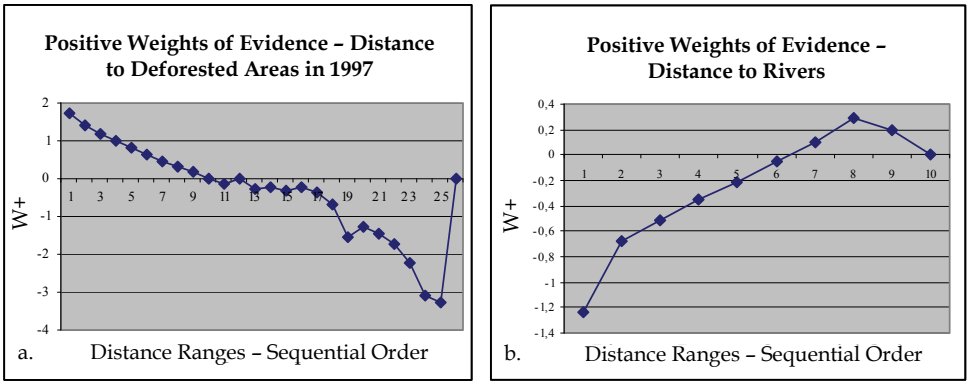


Fig. 7. Behaviour of the positive weights of evidence (W^+) in relation to the distance ranges for: a. distance to previously deforested areas in 1997, and b. distance to rivers

Values of the Positive Weights of Evidence		
Ranges Sequential Order	**Distance to Deforested Areas (m)**	**W^+**
1	0 - 170	1.74594629
2	170 – 240	1.41006378
3	240 - 268	1.18230916
4	268 - 339	0.983065336
5	339 - 360	0.836426843
6	360 - 480	0.650478547
7	480 - 509	0.460166267
8	509 - 537	0.312746731
9	537 – 720	0.179722679
10	720 – 805	-0.0177924833
11	805 – 960	-0.143699617
12	960 – 975	-0.0110620882
13	975 – 988	-0.287776212
14	988 – 1,017	-0.221559314
15	1,017 – 1,045	-0.313774626
16	1,045 – 1,108	-0.247802748
17	1,108 – 1,137	-0.383389447
18	1,137 – 3,325	-0.686325958
19	3,325 – 3,335	-1.54902
20	3,335 – 3,340	-1.25484549
21	3,340 – 3,390	-1.4529907
22	3,390 – 3,757	-1.73329591
23	3,757 – 4,539	-2.2167735
24	4,539 – 4,720	-3.09234848
25	4,720 – 29,889	-3.28531919
26	29,889 – 2,147,483,647	0

Table 2. Values of the positive weights of evidence (W^+) for the distance ranges of the dynamic variable 'distance to previously deforested areas in 1997'

Values of the Positive Weights of Evidence		
Ranges Sequential Order	**Distance to Rivers (m)**	**W^+**
1	0 – 240	-1.23872974
2	240 – 360	-0.681253599
3	360 – 1,320	-0.515391121
4	1,320 – 1,440	-0.353605584
5	1,440 – 25,800	-0.212681384
6	25,800 – 25,920	-0.0495070727
7	25,920 – 29,400	0.104666667
8	29,400 – 29,760	0.286801266
9	29,760 – 32,880	0.193898811
10	32,880 – 2,147,483,647	0

Table 3. Values of the positive weights of evidence (W^+) for the distance ranges of the static variable 'distance to rivers'

Regarding the variable 'distance to rivers' (Fig. 7b), it is possible to observe, however, that the distance ranges closest to rivers present the lowest W^+ values. This can be explained by the fact that, although deforestation processes to a reduced extent occur nearby rivers, these new deforestation patches account for a very limited share of the total deforested area in this region, what causes a decrease in the weights values referring to such distance ranges. As ranges are located ever further from rivers, their weights gradually start to assume increasing positive values. These farthest ranges actually correspond to the very bordering areas of well-established occupations, which are exactly those prone to experience deforestation processes. In brief, this variable acted as a fine tuning device for the variable 'distance to previously deforested areas in 1997'.

As previously stated in Section 4.3, Dinamica EGO generates maps of local probabilities based on the calculated W^+ values, assigning to each map cell a value of transition probability. Fig. 8 presents the map of local land cover change probabilities (from forest to deforested areas), generated in this experiment.

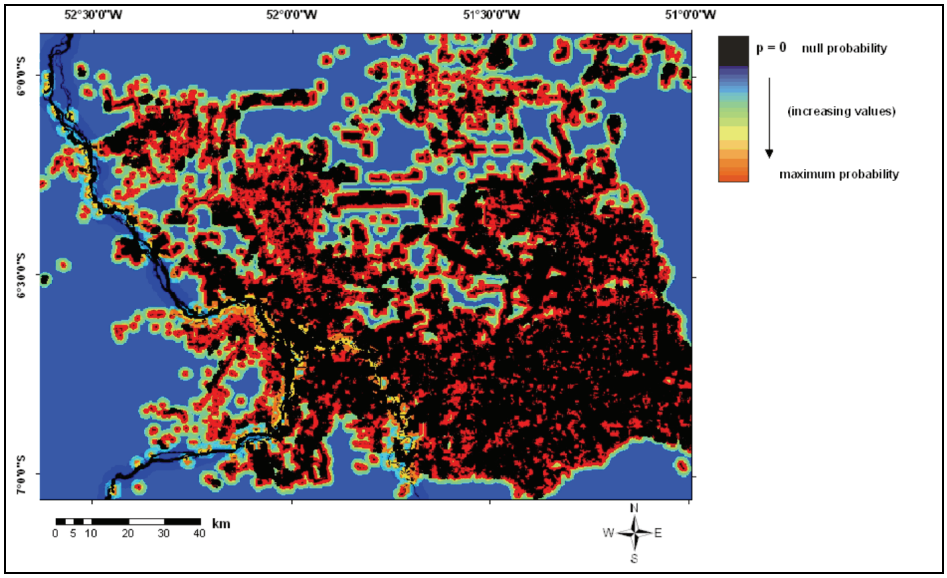


Fig. 8. Map of local land cover change probabilities (from forest to deforested areas) for the period from 1997 to 2000

The areas in black have null values of transition probability. The blueish and greenish cells own very low and low values of probability, while the areas in tones of yellow, orange, and red respectively present intermediate, upper intermediate and high values of transition probability. The areas with the highest probability values pretty much correspond to the areas where deforestation indeed occurred, as it can be observed in the PRODES deforestation map (Fig. 2).

The simulated deforestation map produced by the model (Fig. 9), considering both the static and the dynamic variables, presented high fuzzy similarity indices for multiple spatial resolutions, i.e. for multiple sampling window sizes (Table 4), in the case of the constant decay, what denotes the good performance of the model.

Comparing the simulated land cover map with the real land cover map in 2000 (Fig. 4), it is possible to notice that very regularly-shaped patches generated by real deforestation processes undertaken by capitalized farmers could not be reproduced in the simulation. This can be ascribed to the fact that the current transition functions available in Dinamica EGO still cannot cope with extremely rigid requirements regarding the patches geometry. Nevertheless, the spatial pattern of deforestation patches in the simulated map is very similar to the one found in the real scene, indicating the CA simulation model efficacy. It is worth mentioning that the aim of modelling is not to reproduce reality as close as possible, but solely to detect main spatial patterns and trends of land cover change. Spatial patterns refer to morphological aspects in the patches formation, i.e. whether they are geometrically stable or irregular, whether they are originated by expansion or diffusion processes, and so on, whilst trends concern the directions (or vectors) of deforestation propagation in space.

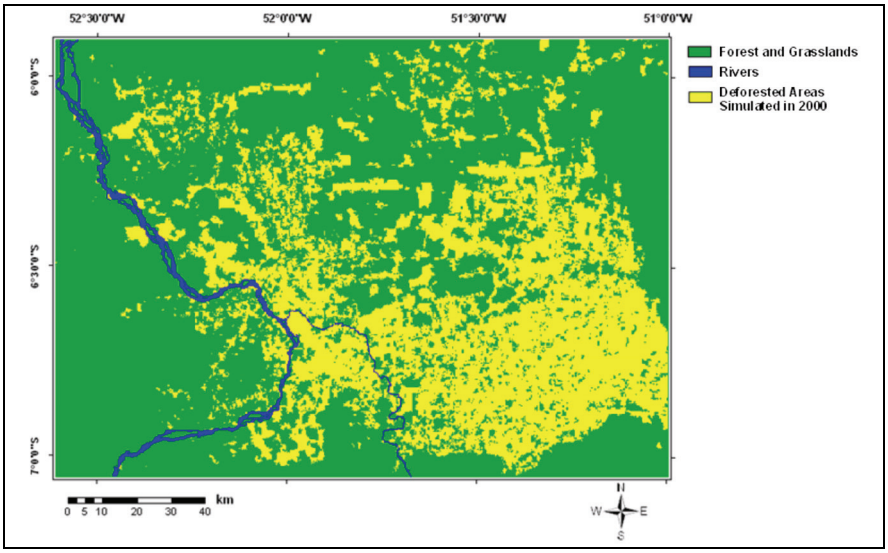


Fig. 9. Map of simulated deforestation for the year 2000, with patches average size set to 300 ha, and variance of size to 500 ha

Since the fuzzy similarity index (FSI) is a flexible method for CA model validation, in the sense that it does not operate on a pixel basis but rather on multiple levels of resolution, the values of this index in the cases where the constant decay is adopted tend to be slightly superior when compared to indices based on strict agreement, which are those derived from a direct pixel-by-pixel comparison between the real and the simulated scene. The agreement increases with the size of the sampling window until it reaches a resolution of around 11 x 11 or 13 x 13 pixels, when the similarity index stabilises (Fig. 10), what shows that the FSI is inefficient to assess the model fitness with rough spatial resolutions. It is worth reminding that the use of agreement indices based on multiple spatial resolutions to assess the performance of CA models can be justified by the fact that it is unfeasible not only to reproduce the configurations of changes in the past, but also and mainly to foresee future

land cover transitions with a very fine spatial accuracy, given the intrinsic randomness of land cover change processes.

CA Deforestation Model	Window Size (Pixels)	Parameters of Patches	FSI* (constant decay)
Simulation from 1997 to 2000	3 x 3		0.877
	5 x 5	μ = 300 ha	0.891
	7 x 7	σ^2 = 500 ha	0.900
	9 x 9	PII** = 1.5	0.903
	11 x 11		0.904
	13 x 13		0.905

*FSI = Fuzzy Similarity Index / PII** = Patch Isometry Index

Table 4. Result of the model validation using different sampling window sizes

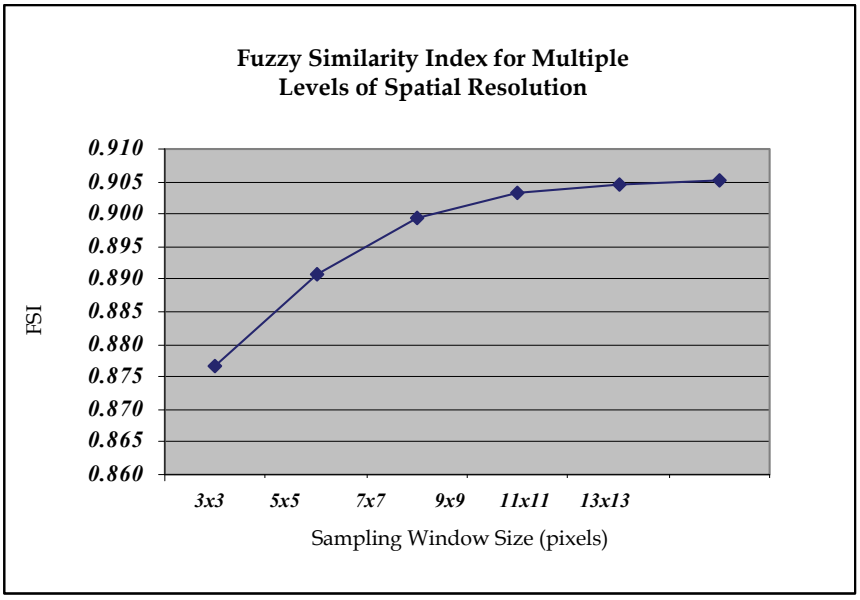


Fig. 10. Variation of the fuzzy similarity index (FSI) as a function of different sampling window sizes

7. Conclusion

This chapter presented an experiment on spatial dynamic modelling based on cellular automata, designed to simulate deforestation in the region of *São Félix do Xingu*, southeast of *Pará* State, north of Brazil, during the period from 1997 to 2000. The employed explaining variables (evidences) do not exhaustively represent the set of deforestation drivers in the area, but they correspond however to strategic inducing factors of such type of land cover change.

The dynamic variable 'distance to previously deforested areas in 1997' was decisive for simulating deforestation processes observed in the vicinities of forest clear-cut areas. In face of the historically reported landed property concentration in this region, the deforestation pattern in the analysed period mostly presents patches of large extensions, which are predominantly found in the surroundings of pioneer areas. The static variable 'distance to rivers', on the contrary, revealed the limited importance of the fluvial transport for deforestation and human occupation in this region. Nevertheless, considering that deforestation still takes place nearby rivers, this variable acted as an alternative for the fine tuning of the variable 'distance to previously deforested areas in 1997'. The obtained result demonstrates the suitability of the employed CA model for simulating deforestation processes in the study area for the time span from 1997 to 2000, what was confirmed by the high values of the fuzzy similarity index.

The Dinamica EGO platform proved to be appropriate for achieving the goals of this research experiment, also in view of one of its transition allocation algorithms – the expander function – which is able to reproduce the spatial pattern of deforestation expansion in the vicinities of previously occupied areas, as it is the case in *São Félix do Xingu*. Further advantages of Dinamica EGO concern its flexible structure that accepts different parameterization methods (Almeida et al, 2008) and varied sets of static and dynamic variables, which, when associated in a thoughtful manner, may meet the modelling requirements of the most diverse study areas, with peculiar land cover (or land use) change patterns. These properties allow the transferability of this and other models developed in this platform to other areas in the Amazon (Soares-Filho et al., 2009; Maeda et al., 2010), owning distinct occupation histories, at different consolidation stages, and involving specific local actors, with very particular characteristics.

As directions for future research, the authors intend to explore a wider scope of explaining variables, like indicators of the land ownership legal status, local unpaved roads as well as vectors of ongoing occupation fronts, besides operating with further stochastic parameterization methods. The authors as well intend to deal with longer temporal series, so as to use a more in-depth knowledge on the area occupation history, and hence, provide differentiated future simulation scenarios for *São Félix do Xingu* in the short- and medium-term. These forecasts would be based on plausible political, socio-economic, and infrastructural arrangements at the local and regional level.

Although models in general have continuously been the target of severe criticism, mainly in view of their reductionism and constraints to fully capture the reality inherent complexity (Briassoulis, 2000), it can be argued that they ought to exist, for they offer an incomparable way of abstracting patterns, order and main dynamic trends of real world processes (Batty, 1976).

Actually, urban models should be conceived, handled, applied and interpreted in a wise and critical way, so that modelers, practitioners, public and private decision-makers as well as citizens as a whole could take the best of what they can offer and sensibly acknowledge their limits (Almeida, 2003; Almeida et al., 2005). Spatial dynamic models, of which CA is one of the best representatives, are still the most promising means for rendering land cover change simulations communicable and transparent to politicians, planners, decision-makers as well as to the lay public in general.

8. Acknowledgement

This work has been accomplished as part of the agenda of the Research Network for Environmental Modelling in the Amazon - GEOMA (http://www.geoma.lncc.br/), established through cooperation agreements among institutions subordinated to the Brazilian Ministry of Science and Technology (MCT).

9. References

Agterberg, F.P. & Bonham-Carter, G.F. (1990). Deriving weights of evidence from geo-science contour maps for the prediction of discrete events, *Proceedings of the XXII International Symposium AP-COM*, pp. 381–395, Berlin, Germany, September 17-21, 1990

Aguiar, A.P.D.; Câmara, G. & Escada, M.I.S. (2007). Spatial statistical analysis of land-use determinants in the Brazilian Amazonia: Exploring intra-regional heterogeneity. *Ecological Modelling*, Vol. 209, No. 02-04, pp. 169–188, ISSN 0304-3800

Almeida, C.M. (2003). Spatial Dynamic Modelling as a Planning Tool: Simulation of Urban Land Use Change in Bauru and Piracicaba (SP), Brazil, In: *Biblioteca do INPE*, 15.09.2010, Available from http://mtc-m12.sid.inpe.br/col/sid.inpe.br/jeferson/2003/12.18.07.36/doc/publicacao.pdf

Almeida, C.M.; Gleriani, J.M.; Castejon, E.F. & Soares-Filho, B.S. (2008). Using neural networks and cellular automata for modeling intra-urban land use dynamics. *International Journal of Geographical Information Science*, Vol. 22, No. 9, pp. 943–963, ISSN 1365-8816

Almeida, C.M.; Monteiro, A.M.V.; Câmara, G.; Soares-Filho, B.S.; Cerqueira, G.C.; Pennachin, C.L. & Batty, M. (2005). GIS and remote sensing as tools for the simulation of urban land use change. *International Journal of Remote Sensing*, Vol. 26, No. 04, pp. 759-774, ISSN 0143-1161

Alves, D.S. (2002). Space-time dynamics of deforestation in Brazilian Amazônia. *International Journal of Remote Sensing,* Vol. 23, No. 14, pp. 2903-2908, ISSN 0143-1161

Amaral, S.; Monteiro, A.M.V.; Câmara, G.; Escada, M.I.S. & Aguiar, A.P.D. (2006). Redes e conectividades na estruturação da frente de ocupação do Xingu-Iriri – Pará. *Geografia*, Vol. 31, No. 3, pp. 655-675, ISSN 0100-7912

Batty, M. (1976). *Urban modelling: algorithms, calibrations, predictions* (1.ed.), Cambridge University Press, ISBN 052-1134-36-6, Cambridge, UK

Becker, B.K. (2005). Geopolítica da Amazônia. *Revista de Estudos Avançados*, Vol.19, No. 53, pp. 71-86, ISSN 0103-4014

Bonham-Carter, G.F. (1994). *Geographic Information Systems for Geoscientists: Modelling with GIS*. Pergamon, ISBN 008-0418-67-8, Ontario, Canada

Brandão Júnior, A.O.; Souza Júnior, C.M.; Ribeiro, J.G.F. & Sales, M.H.R. (2007). Desmatamento e estradas não-oficiais da Amazônia, *Proceedings of the XIII Brazilian Symposium on Remote Sensing*, pp. 2357-2364, ISBN 978-851-7000-31-7, Florianópolis, Santa Catarina, Brazil, April 21-26, 2007

Briassoulis, H. (2000). Analysis of land use change: theoretical and modeling approaches, In: *University of Aegean, Greece*, 08.05.2010, Available from http://www.rri.wvu.edu/WebBook/Briassoulis/contents.htm

Burrough, P.A. (1998). Dynamic modelling and geocomputation, In: *Geocomputation: a primer*, Longley, P.A.; Brooks, S.M.; McDonnell, R. & MacMillan, B., pp. 165-192, John Wiley & Sons, ISBN 047-1985-76-7, Chichester, UK

Costanza, R. (1989). Model goodness of fit: a multiple resolution procedure. *Ecological Modelling*, Vol. 47, No. 03-04, pp. 199-215, ISSN0304-3800

Eastman, J.R. (2003). *IDRISI: The Kilimanjaro edition*, Clark University, Worcester, MA, USA

Escada, M.I.S; Amaral, S.; Monteiro, A.M.V.; Almeida, C.A.; Carriello, F. & Almeida, A. (2007). Padrões de mudança de uso e cobertura da terra na fronteira agropecuária de São Félix do Xingu, PA, *Proceedings of the I Symposium of the Research Network for Environmental Modelling in the Amazon [GEOMA]*, Petrópolis, RJ, Brazil, October 29-31, 2007

Escada, M.I.S.; Vieira, I.C.; Amaral, S.; Araújo, R.; Veiga, J.B.; Aguiar, A.P.D.; Oliveira, I.V.M.; Pereira, J.L.V.; Filho, A.C.; Fearnside, P.M.; Venturieri, A.; Carriello, F.; Thales, M.; Carneiro, T.S.G.; Monteiro, A.M.V. & Câmara, G. (2005). Processos de ocupação nas novas fronteiras da Amazônia (o interflúvio do Xingu/Iriri). *Estudos Avançados*, Vol. 19, No. 54, pp. 9-23, ISSN 0103-4014

Research Network for Environmental Modelling in the Amazon [GEOMA]. (2004). *Dinâmica territorial da frente de ocupação de São Félix do Xingu-Iriri: Subsídios para o desenho de políticas emergenciais de contenção do desmatamento*, Ministério da Ciência e Tecnologia, Brasília, Brazil

Goodacre, A.K.; Bonham-Carter, G.F.; Agterberg, F.P. & Wright, D.F. (1993). A statistical analysis of the spatial association of seismicity with drainage and magnetic anomalies in western Quebec. *Tectonophysics*, Vol. 217, No. 03-04 , pp. 285-305, ISSN 0040-1951

Hagen, A. (2002). Multi-method assessment of map similarity, *Proceedings of the 5th AGILE Conference on Geographic Information Science*, pp. 171–182, Palma de Mallorca, Spain, April 25-27, 2003

Hagen, A. (2003). Fuzzy set approach to assessing similarity of categorical maps. *International Journal of Geographical Information Science*, Vol. 17, No. 3, pp. 235-249, ISSN 1365-8816

Hagen-Zanker, A.; Straatman, B. & Uljee, I. (2005). Further developments of a fuzzy set map comparison approach. *International Journal of Geographical Information Science*, Vol. 19, No. 7, pp. 769-785, ISSN 1365-8816

Hagen-Zanker, A. (2006). Map comparison methods that simultaneously address overlap and structure. *Journal of Geographical Systems*, Vol. 8, No. 2, pp. 165-185, ISSN 1435-5930

Instituto Nacional de Pesquisas Espaciais [INPE]. (2006). PRODES, In: *OBT-INPE*, 28.11.2007, Available from http:// http://www.obt.inpe.br/prodes/

Laurance, W.F.; Albernaz, A.K.M.; Schroth, G.; Fearnside, P.M.; Bergen, S.; Venticinque, E.M. & Costa, C. (2002). Predictors of deforestation in the Brazilian Amazon. *Journal of Biogeography*, Vol. 29, No. 05-06, pp. 737-748, ISSN 1365-2699

Maeda, E.E.; Almeida, C.M.; Ximenes, A.C.; Formaggio, A.R.; Shimabukuro, Y.E.; Pellikka, P. (2010). Dynamic modeling of forest conversion: Simulation of past and future scenarios of rural activities expansion in the fringes of the Xingu National Park, Brazilian Amazon. *International Journal of Applied Earth Observation and Geoinformation*, ISSN 0303-2434, In press, DOI 10.1016/j.jag.2010.09.008

Openshaw, S. (2000). Geocomputation, In: *Geocomputation,* Openshaw, S.; Abrahart, R. J., (Eds.), pp. 1-31, Taylor & Francis, ISBN 978-047-1985-76-1, New York, USA

Pereira, L.M.; Escada, M.I.S. & Rennó, C.D. (2007). Análise da evolução do desmatamento em áreas de pequenas, médias e grandes propriedades na região centro-norte de Rondônia, entre 1985 e 2000, *Proceedings of the XII Brazilian Symposium on Remote Sensing*, pp. 6905-6912, ISBN 851-7000-18-8, Goiânia, GO, Brazil, April 16-21, 2005

Pontius Jr., G. (2002). Statistical methods to partition effects of quantity and location during comparison of categorical maps at multiple resolutions. *Photogrammetric Engineering & Remote Sensing*, Vol. 68, No. 10, pp. 1041-1049, ISSN 0099-1112

Rodrigues, H.O.; Soares-Filho, B.S. & Costa, W.L.S. (2007). Dinamica EGO, uma plataforma para modelagem de sistemas ambientais, *Proceedings of the XIII Brazilian Symposium on Remote Sensing*, pp. 3089-3096, ISBN 978-851-7000-31-7, Florianópolis, Santa Catarina, Brazil, April 21-26, 2007

Santana, L.F. (2007). *São Félix do Xingu e sua história: 1889 – 1997*, Prefeitura Municipal de São Félix do Xingu, São Félix do Xingu

Schimink, M. & Wood, C.H. (1992). *Contested Frontiers in Amazonia*, Columbia University Press, New York, USA

Soares-Filho, B.S.; Assunção, R.M. & Pantuzzo, A.E. (2001). Modeling the spatial transition probabilities of landscape dynamics in an Amazonian colonization frontier. *BioScience*, Vol. 51, No. 12, pp. 1059-1067, ISSN 0006-3568

Soares-Filho, B.S.; Cerqueira, G.C. & Pennachin, C.L. (2002). DINAMICA – a stochastic cellular automata model designed to simulate the landscape dynamics in an Amazonian colonization frontier. *Ecological Modelling*, Vol. 154, No. 03, pp. 217-235, ISSN 0304-3800

Soares-Filho, B.S.; Nepstad, D.C.; Curran, L.M.; Cerqueira, G.C.; Garcia, R.A.; Ramos, C.A.; Voll, E.; McDonald, A.; Lefebvre, P. & Schlesinger, P. (2006). Modelling conservation in the Amazon basin. *Nature*, Vol. 440, No. 7083, pp. 520-523, ISSN 0028-0836

Soares-Filho, B.S.; Rodrigues, H.O. & Costa, W.L.S. (2009). Modeling Environmental Dynamics with Dinamica EGO, In: *CSR-UFMG*, 23.06.2008, Available from http://www.csr.ufmg.br/dinamica/, ISBN 978-85- 885910119r-r0-2

Wolfram, S. (1983). Statistical mechanics of cellular automata. *Review of Modern Physics,* Vol. 55, pp. 601-643, ISSN 0034-6861

4

Spatial Optimization and Resource Allocation in a Cellular Automata Framework

Epaminondas Sidiropoulos and Dimitrios Fotakis
Faculty of Engineering, Aristotle University of Thessaloniki
Greece

1. Introduction

Land management is a complex activity associated with the determination of land uses, the placement of activities and facilities and the distribution of resources over extensive territories with a view to satisfying one or more criteria of economical and/or ecological character. It follows from this descriptive definition that in land management it is not sufficient to distribute goods or commodities to a number of beneficiaries, but, mainly, to carry out planning with respect to space and location, thus intervening and shaping the local geography of economical and environmental characteristics.

An important part of land management is land use planning, in which a given area is divided into land blocks with each one of them being assigned a specific land use, taken from a set of possible land uses. The search for suitable combinations of land uses, so as to attain given objectives, constitutes a computationally intensive optimization problem. A related problem concerns spatial resource allocation, in which one or several resources have to be allocated to each one of the described land blocks, again in order to attain preset objectives and possibly satisfy constraints. The sought for distribution and nature of these resources bears a strong relation to the land uses of the respective blocks. This fact gives rise to even more difficult, but also more realistic optimization problems.

A basic resource to be managed is water. Allocating water may not simply involve its unit price, but also the estimation of transportation and extraction costs. In the latter case physical modeling of groundwater movement and pumping is needed and this contributes to the complexity and nonlinearity of the problem. This fact makes the present problem different from the typical allocation problems. Problems of land use planning and water allocation combined with water extraction have been presented by Sidiropoulos & Fotakis (2009) and Fotakis (2009) and are reviewed in this chapter, along with new results concerning a cellular – genetic approach.

Genetic algorithms and cellular automata will be the basic tools to be implemented in the present approach. Genetic algorithms are well-known biologically-inspired meta-heuristics. Their properties and characteristics are described in textbooks such as Michalewicz (1992) and Goldberg (1989). Applications abound in the literature.

Cellular automata date back to von Neumann. Their fundamental importance was demonstrated by Wolfram (2002). They have been used as a background for modeling a great diversity of natural, as well as social and economic systems. Cellular automata have been used for simulating natural phenomena. Also, numerous applications have been

presented for the spatial analysis of ecosystems. Hogeweg (1988) used them to simulate changes in landscape. Green et al (1985), Karafyllidis and Thanailakis (1997), Karafyllidis (2004) and Supratid & Sunanda (2004) employed cellular automata to simulate the spread of fire in a forest ecosystem, while Sole and Manrubia (1995) simulated the dynamics of forest openings by means of cellular automata. Also, cellular automata have been used for the simulation of succession and spatial analysis of vegetation growth (Colasanti and Grime, 1993). A useful application of cellular automata concerns the study of spatial characteristics of the socio-economic development. Balmann (1997) analyzed the structural change in a rural landscape with the help of a two-dimensional cellular automaton, and Deadman et al (1993) used cellular automata to model the development of rural settlements, while Jennerette and Wu (2001) used them to model urban development, along with producing possible land-use scenaria. Finally, Prasad (1988) described the economy as a cellular automaton where the self-organization responds to the evolution of the system seeking a more efficient provision of social resources.

Recently, cellular automata are used for various optimization problems such as finding the optimal path (Adamatzky 1996), designing of sewerage systems (Guo etc., 2007), management of groundwater aquifers (Sidiropoulos and Tolikas, 2008), reservoir management (Afshar and Shahidi, 2009) and optimization of forest planning (Strange et al. 2001, Heinonen and Pukkala 2007, Mathey et al. 2008) and afforestation (Strange et al. 2002).

The use of evolutionary methods in spatial optimization problems such as the ones outlined here is called for by their complexity and nonlinearity. Additionally, a particular characteristic of these problems is the relation between local interactions and global system behavior. These considerations lead to the introduction of cellular automata.

The area under study may be modeled as a cellular automaton with the land blocks being represented by the cells of the automaton. The actual spatial arrangement of the land blocks provides the neighborhood structure of the cells. The states of each cell represent the land uses or the water sources corresponding to the land block represented by the particular cell.

In the typical cellular automaton, a transition rule is required operating on each cell as a function of the states of the cell and of its neighbors. In the literature such rules have been determined in order to construct cellular automata that perform certain computational tasks (Mitchel et al., 1994 and many subsequent reports along the same basic idea). In the present approach no constant rule is sought. Instead, genetic algorithms will be embedded into the cellular automaton in order to guide its evolution. More specifically, two types of genetic algorithm will be implemented:

1. The operative genetic algorithm, which will indicate each time a replacement rule for each block. No constant rule will be sought and no decomposition of the objective function will be involved.
2. The natural genetic algorithm endowed with a neighborhood rule. This rule will operate on a neighborhood level and on the basis of local values of the objective function for the purpose of enhancing the performance of the natural genetic algorithm.

Both these approaches have been presented in different publications (Sidiropoulos and Fotakis, 2009; Fotakis, 2009) but have not been compared or combined. This is done by their juxtaposition in the present chapter.

The natural genetic algorithm works on the whole configuration and its genetic operators are not based on local interactions among neighboring cells. Therefore, despite the cellular background, it would not by itself qualify as a cellular – genetic scheme. The addition of the

neighborhood rule introduces the local element into the computational scheme and, therefore, it brings the whole scheme closer to the cellular prototype.
The operative genetic algorithm, on the other hand, is fully consistent with the cellular automaton model. This algorithm defines each time a renewed rule for synchronous changes to each cell on the basis of the neighboring states.
A central issue in spatial optimization is the balance of the local versus the global aspects of the problem. The introduction of the above local rules serves the purpose of guiding iteratively the whole cellular automaton to optimal conditions. But the definition of such rules is not self-evident. The objective function of the problem generally depends on the entire configuration and decomposition or reduction to cell contributions is not evident or even feasible. Both schemes deal with this issue, which is also discussed in relation to the relevant literature.
In both genetic-cellular schemes the resulting arrangements of the cellular automaton present greater compactness, in comparison to the natural genetic algorithm, with respect to subareas with the same land use or water source. This is a significant qualitative result for land management (Vanegas et al, 2010). Moreover, in the present approach this characteristic is obtained as an emergent result without the addition of any special constraints or the modification of the objective function, as in similar problems of the literature.

2. Description and formulation of the problem

A hypothetical problem is considered in order to illustrate land management combined with groundwater allocation (Sidiropoulos and Fotakis, 2009). The terrain is represented in the form of a two-dimensional grid, the nodes of which correspond to land blocks.
The resource to be allocated is water and the cost involved will consist of two parts related to extraction and transport, respectively. Each one of the blocks is connected to one of the wells. These connections are the decision variables of the problem. They can be depicted as in Figure 1. The color of each cell indicates the well to which it is connected.
The transport cost for each block is taken as proportional to the distance of the block from the respective well. The total transport cost results from summing over all land blocks. The pumping cost is estimated via a steady-state groundwater phreatic aquifer model. The sum of the two costs forms the objective function to be minimized.

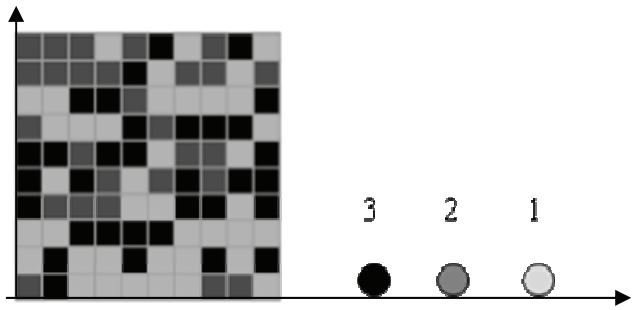


Fig. 1. Two-dimensional mosaic with three wells

Let (i, j) be the coordinates of the center of the typical block with i= 1,2,…a and j=1,2,..,b, where a and b are the lengths of the two sides of the orthogonal grid. The blocks can be numbered consecutively, row by row. If $k = 1, 2, ..., a \bullet b$, then

$$i = k - a\left[\frac{k}{a}\right], \qquad j = \left[\frac{k}{a}\right] + 1 \tag{1}$$

where the brackets denote the integer part of the enclosed number.
Let m be the number of the wells and let the wells be numbered from 1 to m. Also, let $w_k \in \{1, 2, .., m\}$ be the number of the well assigned to block k (Fig.2) with $k = 1, 2, ..., a \bullet b$, according to the above numbering.

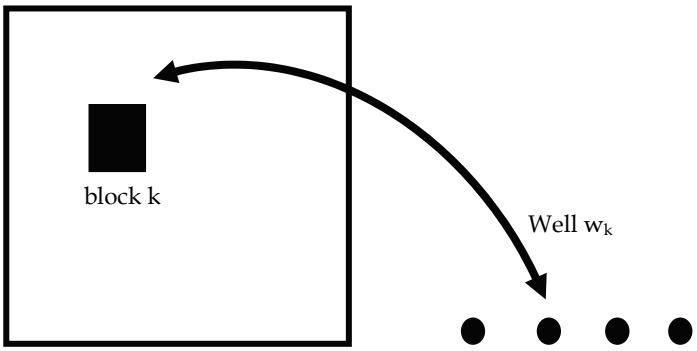


Fig. 2. Correspondence of cell to well

Then the transport cost is

$$F_T = \sum_{k=1}^{a \bullet b} \left[\left(x_k - x_{w_k} \right)^2 + \left(y_k - y_{w_k} \right)^2 \right]^{1/2} \tag{2}$$

where (x_k, y_k) are the coordinates of block k and (x_{w_k}, y_{w_k}) the coordinates of the respective well.
The pumping cost is expressed as follows:
Let $s_w, w \in \{1, ..., m\}$ be the number of blocks irrigated from well w. Then the discharge from well w is equal to

$$Q_w = \sum_{k=1}^{s_w} q_k \tag{3}$$

where α_k is a quantity representing the water needs of block k.
The drawdown at each well is given by

$$\Delta h_w = \frac{1}{2\pi b} \sum_{\substack{w'=1 \\ w' \neq w}}^{m} \frac{Q_{w'}}{k_{w'}} \ln \frac{\sqrt{(x_w - x_{w'})^2 + (y_w - y_{w'})^2}}{R} + \frac{Q_w}{2\pi b k_w} \ln \frac{r_w}{R} \tag{4}$$

where b is the thickness of the aquifer, assumed constant, R is the influence radius, r_w is the radius of well w and k_w with w=1,2,..,m are the hydraulic conductivities of the areas around each one of the wells.
Finally, the total pumping cost is proportional to the quantity

$$F_P = \sum_{w=1}^{m} Q_w \Delta h_w \tag{5}$$

where Q_w and Δh_w are given by Equations (3) and (4).
Thus, from Equations (2) and (5), the total cost can be taken to be equal to the sum

$$F=F_T+F_P \tag{6}$$

From the above formulation it can be seen that the present problem differs from classical resource allocation problems, because the cost associated to each cell does not depend only on the quantity of the water to be supplied to the particular block. It also depends on the position of the block itself. In fact this is true both for the transport cost, which depends on distances, and for the pumping cost, which is determined through the aquifer model with its predominantly spatial character. Moreover, the most distinct difference comes from the fact that the pumping cost is influenced not only by the well connected to the particular block, but also by the action of the other wells.
Questions of spatial decomposition will be addressed in a subsequent section, along with a comparison of the present problem to typical problems of spatial resource allocation, as they appear in the relevant literature. It needs to be noted that typical problems are based on separability of the individual cell contributions, which does not hold true for the present problem.
The solution methods to be described in the following sections are specially adjusted to the spatial – cellular character of the problem domain. The present approach is further compared to the treatment of various types of resource allocation problems of the recent literature, in particular to problems of forest planning.

3. Method of solution

3.1 A natural genetic algorithm

The objective function, Equation (6), is to be minimized as a function of the connections of the various blocks to the water wells. The objective function is nonlinear and the problem is one of combinatorial optimization.
Evolutionary algorithms are particularly suited for this kind of resource allocation problem. Indeed, such methods have been applied to resource allocation problems (Khan et al. 2008, Magalhaes-Mendes 2008), as well as problems involving both water allocation and crop planning (Ortega et al. 2004, Zhou et al. 2007, Khan et al. 2008).
A natural way of encoding the problem in a genetic algorithm framework has been presented by Sidiropoulos and Fotakis (2009). Here, it is cast into a more general framework. Let L be the set of cells as numbered according to the above scheme:

$$L = \{1,2,.., a.b\}$$

and W the set of the m wells numbered from 1 to m:

$$W = \{1,2,\ldots,m\}.$$

Let $p : L \to W$ be a function assigning a well to each cell. This function gives rise to a set

$$C = p(L) \equiv \{p(1), p(2), \ldots, p(a \cdot b)\} = \{w_1, w_2, \ldots, w_{a \cdot b}\} \tag{7}$$

where $w_k = p(k) \in W, \quad k = 1,2,\ldots,a \cdot b$

The set C is called a configuration of the cellular automaton.
Consider N such functions and let

$$C_i = p_i(L), \quad i = 1,2,\ldots,N$$

The C_i's represent possible configurations and they can be taken as the chromosomes of the natural genetic algorithm, with each chromosome expressing a distribution of the water sources among the cells.
The genetic operators of selection, crossover and mutation (Michalewicz, 1992) are applied to the above population of the N chromosomes. In particular, the crossover operator follows a two-dimensional pattern (e.g. Moon et al, 1997; Sidiropoulos and Fotakis, 2009), as shown in Fig. 3

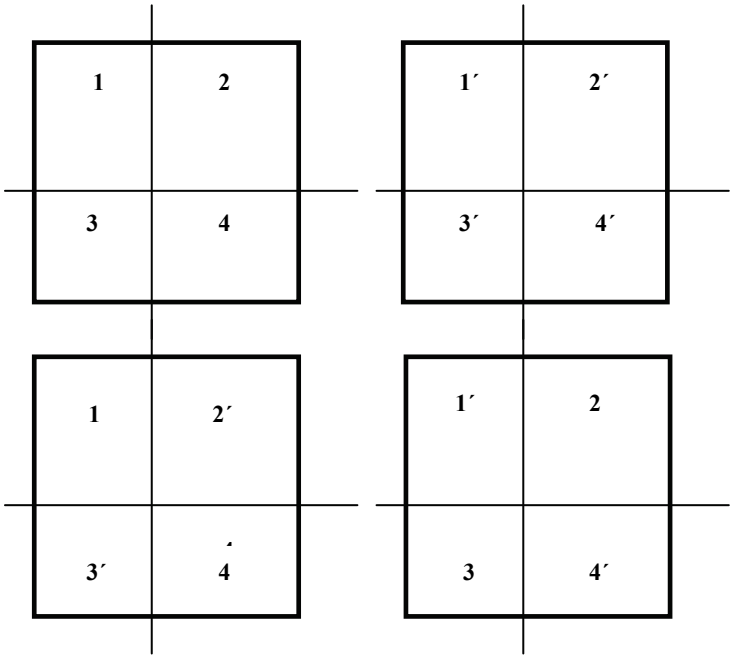


Fig. 3. Parents (above) and offspring (below)

3.2 An operative genetic approach

The alternative scheme is based on the notion of neighborhood and is thus consistent with the cellular automaton approach. A neighborhood is assigned to each block. The

neighborhood is defined in the sense of von Neumann (east – west and north – south cells, e.g. Gaylor and Nishidate,1996).
Let N_k be the neighborhood of cell k and let $n_k \in N_k$ be functionally related to k: $k \mapsto n_k$. Then L is transformed to

$$L_n = \{n_1, n_2, ..., n_{a \cdot b}\} \tag{8}$$

and a new configuration can be defined as

$$\bar{C} = \{\bar{w}_1, \bar{w}_2, ..., \bar{w}_{a \cdot b}\} \tag{9}$$

where $\bar{w}_k = p(n_k)$, as prescribed by Equation (7).
The above can be written more succinctly in operative form:
Let $q : L \to L$ be the above function relating k to n_k such that

$$q(k) \in N_k \quad \forall k \in L .$$

Then

$$n_k = q(k), \; L_n = q(L) \text{ and } \bar{w}_k = p(q(k)) .$$

Thus an operator O can be defined as

$$O[p(k)]=p(q(k)) \text{ and } \bar{C} = O[C]$$

where $\bar{C}$ is the transformed configuration
Therefore, starting from a configuration

$$C = \{w_k \mid w_k \in W \quad and \quad k = 1,2,..,a \bullet b\}$$

the transformed configuration

$$\bar{C} = C_n = \{w_{n_1}, w_{n_2}, ..., w_{n_{a \cdot b}}\} \tag{9}$$

is obtained via the operator O. The list L_n of Equation (8) induces the function q(k) and thus it generates the operator O. L_n will become the typical chromosome:

$$L_n = \{n_k \mid n_k \in N_k \quad and \quad k = 1,2,..,a \bullet b\} \tag{10}$$

Considering N functions q_i, i=1,2,..,N, like the function q, N corresponding operators O_i result with N new transformed $\bar{C}_i$ configurations.
The generating lists L_{ni}, i=1,2,…N constitute the chromosome population in the operative genetic algorithm.
Also, according to this notation, let $w_{n_k} \in \{1,2,..,m\}$ be the number of the well assigned to the neighbor cell n_k.
The chromosome L_{ni} of Equation (10) can be considered as a replacement rule that dictates to the block k to replace the well w_k assigned to it by the the well w_{nk} of the selected neighboring block n_k (Fig.2).
The appropriate choice of the suitable neighbor n_k will be put forward by the genetic algorithm.

The algorithm can be summarized as follows:

a. An initial reference configuration is formed by taking w_k = Random(1,m) for k = 1,2,..,a.b.
b. An initial population is created consisting of chromosome operators of the type (10).
c. The operators act on the reference configuration resulting in N new configurations emanating from these N transformations of the reference configuration.
d. The N configurations of (c) are evaluated and the best one becomes the new reference configuration.
e. Stopping criteria are applied on best configurations obtained up to the current generation
f. The evaluations of (d) are assigned to the chromosomes of (b) and the operations of selection, crossover and mutation are applied on them and a new population of operators results. Control is transferred to step (c).

The algorithm is shown schematically in Fig. 4.

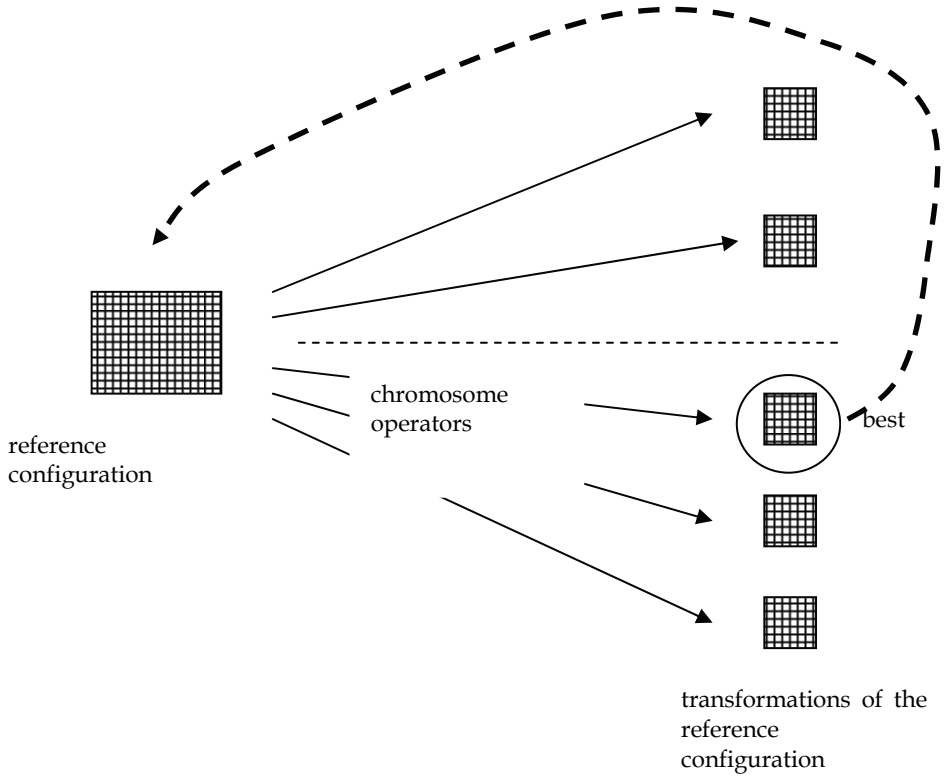


Fig. 4. Schematic representation of the cell-based operative genetic algorithm

4. Local versus global objectives

A basic issue in spatial optimization concerns the formulation of local objectives in relation to the overall global objectives, as the latter are expressed through the objective function. Local objectives permit the design of local transition rules in cellular automata. The optimization problem presented above was treated by means of operators applied to the individual cells in the local sense without decomposing the objective function into local contributions. Another approach would be to define local components of the objectives and then attempt to reduce the overall problem to the solution of the partial corresponding problems at the neighborhood level of each cell. In the literature various approaches can be noted in relation to this issue.

Strange et al (2001) were among the first researchers to employ cellular automata concepts in spatial optimization. They considered individual cell contributions of the objective function. However, some of these contributions, although referred to a particular cell, relate to the wider configuration as they express neighborhood properties. Optimization is performed on the basis of optimizing each one of these cell components.

In Heinonen and Pukkala (2007) the objectives are distinguished into local and global ones and a composite objective function is used consisting of the weighted sum of a local and a global term. The weighting coefficient of the global term is gradually increased in the course of a successive local and global solution of the optimization problem. The local part is treated by means of updating, via mutations, the cells of an underlying automaton.

Seppelt and Voinov (2002) in a grid search approach present a clear distinction between a local and a global method. Their objective function consists of a sum of cell-dependent terms and the solution consists of two stages, a local and a global one. The latter is performed through a genetic algorithm. Although a grid forms the basis of the problem formulation, no complete cellular automaton characteristics, such as local transition, appear in the whole treatment.

Aerts and Heuvelink (2002) and Aerts et al. (2005) again represent their studied area in the form of a grid of cells. They employ simulated annealing and genetic algorithms for the spatial optimization problem. No cellular automaton procedure can be discerned, although their special crossover operator is designed in a way that prevents fragmentation of the grid domain. Various criteria for compactness and contiguity are considered by suitably augmenting the objective function.

Specially designed operators are also presented by Datta et al (2007) in order to favor more compact configurations.

Ligmann-Zielinska et al.(2008) describe the SMOLA (Sustainable Multiobjective Land Use Allocation) software. The spatial optimization problems treated under this software belong to the class of linear and integer programming. Contiguity and compactness are handled by means of separate constraints.

The present problem differs from the typical problems of spatial analysis by the fact that a large part of the computational effort is placed on the allocation of the resource and on modeling its extraction and transport. Thus optimization has to be achieved both in terms of the spatial arrangement and with respect to water extraction efficiency. It is worth noting that there are two points of view, one emphasizing spatial optimization and treating necessary resources as optimization parameters and the other one emphasizing resource management and regulating spatial arrangements toward economy in resource spending. The latter approach may be encountered in the hydraulic engineering literature.

Characteristically, Yeo and Guldmann (2010) investigate land use allocation with a view to minimize watershed peak runoff. Similarly, the same authors (Yeo et al., 2004) optimize land use patterns to reduce both peak runoff and nonpoint source pollution. Notably, Sadeghi et al (2009) optimize land uses with a view both to reducing soil erosion and maximizing benefit. However, their problem is formulated in terms of linear programming.
In general, the objective functions of these composite problems are both nonlinear and non-separable with respect to partial cell contributions. However, local objective functions are defined in the present approach without a strictly local character. These local functions will not supplant the objective function for the optimization procedure. They will only be used for the formulation of an auxiliary operator, called the neighborhood rule (Fotakis, 2009, Fotakis and Sidiropoulos, 2010). This operator falls into the category of learning operators (Hinton & Nowlan, 1987, Krzanowski & Raper, 2001). These operators are characterized by their action on the phenotype and not on the genetic composition of the objects under study. Learning operators have been demonstrated to facilitate the evolutionary process.
The definitions of local objectives for the problem under study are given in the next section.

4.1 Local objectives

The transport cost F_T (Equation 2) can be decomposed into cell contributions as follows:

$$f_{Tk} = [(x_k - x_{wk})^2 + (y_k - y_{wk})^2]^{1/2} \tag{11}$$

for k=1,2,…,a.b.
The pumping cost F_P (Equation 5) does not lend itself to such a direct decomposition. Let q_k denote the discharge corresponding to the given water needs of block k. The subscript j indicates that this block is connected to well j.
Let S_j be a subset of the set L of the cells, characterized by the fact that its members are connected to well j. Because each one of the cells is connected to one well only,

$$\bigcup_{j=1}^{m} S_j = L, \qquad S_i \bigcap S_j = \varnothing \;\; i \neq j, \;\; i, j = 1,2,..,m$$

Then the discharge of well j is equal to

$$Q_j = \sum_{k \in S_j} q_k \tag{12}$$

It follows from Equation (4) that the drawdown at the position of well j, j=1,2,..,m. is a nonlinear function of $Q_1, Q_2,\ldots,Q_m$.
The pumping cost F_P , from Equation 5 and Fig. 2 relating k to w_k, can be rewritten as

$$F_P = \sum_{j=1}^{m} Q_j \Delta h_j = \sum_{j=1}^{m} \left(\sum_{k \in S_j} q_k \right) \Delta h_j = \sum_{k=1}^{a \cdot b} q_k \Delta h_{w_k} \tag{13}$$

The local value of the pumping cost can now be written as follows based on equation (13):

$$f_{Pk} = q_k \Delta h_{w_k}, \;\; k = 1,2,...,a \cdot b \tag{14}$$

It must be noted again that Equation 14 does not represent a strict separation into cell contributions. However this expression will be used beneficially in the local sense, as it will be shown below.
The local objective function may now be defined as

$$f_k = f_{Tk} + f_{Pk}, \quad k = 1,2,..,a \cdot b \tag{15}$$

The nature of the above decomposition will now be examined more closely.

4.2 Decomposition and deviation from the classical resource allocation problem

For the typical resource allocation problem the following basic formulation, with certain variations, is given in the pertinent literature (Aerts & Heuveling, 2002, Aerts et al, 2008, Datta et al, 2007, Lingmann – Zielinska et al, 2010):

Minimize $$F = \sum_i \sum_j \sum_\ell C_{i,j,\ell} x_{ij\ell} \tag{16}$$

where $C_{i,j,\ell}$ is the cost associated with the land use ℓ and

$$x_{ij\ell} = \begin{cases} 1 & if \quad block \quad i,j \quad is \quad assigned \quad land \quad use \quad \ell \\ 0 & otherwise \end{cases}$$

The subscripts i and j run over the coordinates of all cells and the subscript ℓ runs over all possible land uses.
In the literature a number of constraints is added to the above formulation expressing e.g. allowable percentages in the distribution of land uses, demands on compactness, restrictions on fragmentation etc.
The above expression (16) may assume the following equivalent and somewhat simpler form under the present notation:

Minimize $$F = \sum_k \sum_\ell C_{k,\ell} x_{k\ell} \tag{17}$$

and $$x_{ij\ell} = \begin{cases} 1 & if \quad block \quad k \quad is \quad assigned \quad land \quad use \quad \ell \\ 0 & otherwise \end{cases}$$

where now k runs over all cells under the numbering introduced in section 2.
Typically, $C_{k\ell}$ depends solely on the cell's position and on the land use attached to it. In that case the problem can be solved by linear programming, provided the constraints are also linear.
Expression (17) for the objective function can be rewritten as follows, in order to allow comparisons with the present problem formulation:

$$F = \sum_{\ell=1}^{m} \sum_{k \in S_\ell} C_{k,\ell} \tag{18}$$

where S_ℓ is the set of cells assigned to the land use ℓ and m is the number of possible land uses.

In the present problem the land uses are identified with the water sources and expression (18) can be compared to equation (13), which gives the objective function for the pumping cost, i.e.

$$C_{k,\ell} = q_k \Delta h_{\ell_k} \tag{19}$$

It is important to note here that the coefficient $C_{k\ell}$ does not depend solely on the land use of cell k. Indeed, by virtue of Equation (4), Δh_{ℓ_k} depends on the discharges of all the wells and the discharges again are determined by the distribution of the wells among the cells. Therefore, $C_{k\ell}$ depends on the land uses of the other cells as well as on the land use of cell k. This fact differentiates the present problem from the typical case of the spatial optimization literature. The cell decomposition discussed above is utilized in order to define a local, neighborhood operator.

4.3 Neighborhood rule

An operator acting on the neighborhood of each cell is called the neighborhood rule and is defined as follows:

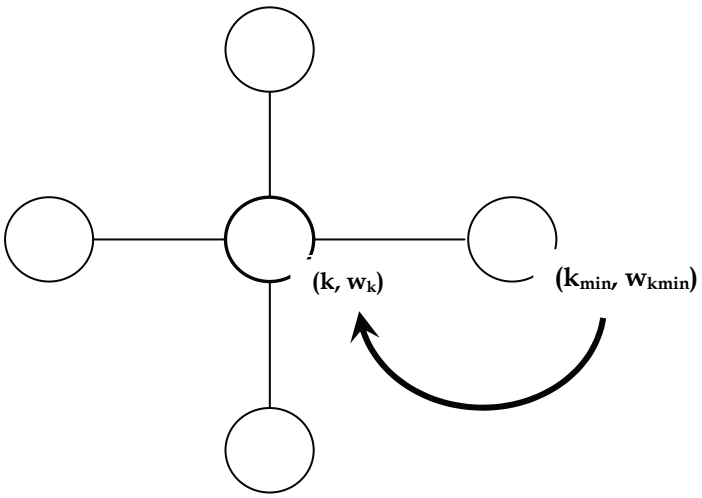


Fig. 5. Neighborhood rule

Let N_k be the neighborhood of cell k, consisting of l elements:

$$N_k = \{k_1, k_2, ..., k_l\}$$

For each one of the neighborhood elements the local objective function (15) is evaluated. Let k_{min} be the element of N_k with the minimum value of the local objective function:

$$f_{k_{\min}} = \min\{f_{k_1}, f_{k_2}, ..., f_{k_l}\} .$$

Then the state of the current cell k is replaced by the state of the state of the cell k_{min} (Fig.5):

$$w_k \leftarrow w_{k_{\min}} .$$

The operation just described may be considered as a kind of mutation applied on the phenotype. It is applied with a certain probability to all genes (cells) of all chromosomes.

4.4 Neighborhood mutation

Another operator that does not depend on the local objective functions and acts on the phenotype is now introduced under the name neighborhood mutation. It differs from the classical version by the fact that it is tied to the neighborhood of the current gene-cell. Thus it is also consistent with the notion of the cellular automaton.

More specifically, for every cell a random number r is taken between 0 and 1 and if $r < p_{nm}$, where p_{nm} is the preset neighborhood mutation probability, then the state of the cell is replaced by the state of one of its neighbors. The latter is again chosen at random (Fig.6).

In the notation of the previous section,

For each k (k=1,2,…,a.b)

if $r < p_{nm}$,

let s be a random integer between 1 and l and replace

$$w_k \leftarrow w_{k_s} .$$

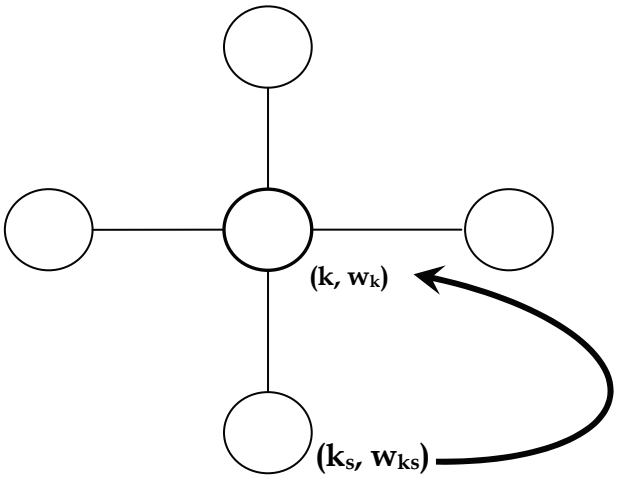


Fig. 6. Neighborhood mutation

Both of the above kinds of mutation will be tried and the results will be presented in the next section.

5. Results

A fictive spatial optimization problem (Sidiropoulos & Fotakis, 2009) is formulated in order to illustrate the methods and operators presented here. The area under study is represented as a 10x10 grid with the wells placed in the positions shown in Fig.1. The wells' coordinates are $x_{w1}=20$, $y_{w1}=0$, $x_{w2}=18$, $y_{w2}=0$, $x_{w3}=15$, $y_{w3}=0$. The hydraulic conductivities around wells

1, 2, 3 are k_1=0.05x10-3m/s, k_2=0.5x10-3m/s, k_3=1.2x10-3m/s, respectively. The constant thickness of the underlying aquifer is b=50m, the radius of influence R=15m and the radii of the wells all equal to r_w=0.10m.

The natural GA was first compared to the operative GA, as in Sidiropoulos & Fotakis (2009). The result obtained by the operative GA is clearly superior to the one of the natural GA both in terms of the objective function value and in terms of compactness (Figs. 7 and 8). The arrangement depicted in Figure 8 was reached within 160 generations and stabilized thereafter, while the arrangement of Fig. 7 was the best result obtained by the natural GA within 1000 generations.

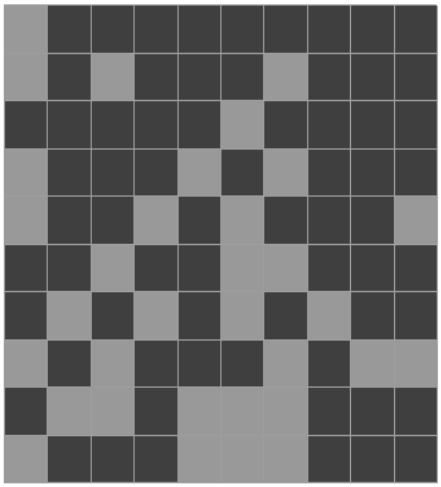


Fig. 7. Natural genetic algorithm

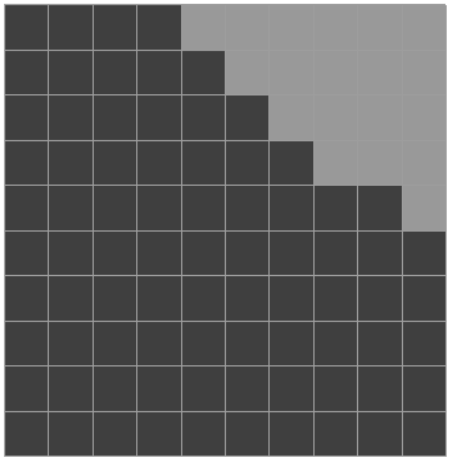


Fig. 8. Operative genetic algorithm

Subsequently, the natural GA was enhanced by the addition of the neighborhood rule (NR) described in section 4.3. The arrangement shown in Figure 9 was obtained within 600 generations. It presents a better picture in terms of compactness compared with the natural GA (Fig. 7), but not as good as the arrangement of the operative GA (Fig. 8). However, the objective function value attained in this case is even better than the one given by the operative GA.

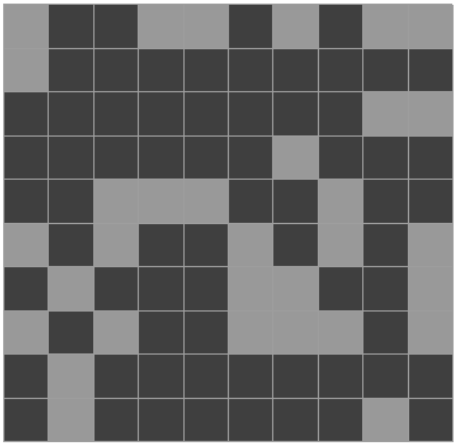


Fig. 9. Natural GA with NR

In a similar fashion the natural GA was combined with the neighborhood mutation (NM) of section 4.4. The result is shown in Figure 10. Concerning compactness it is clearly better than the one of Figure 9 and almost as good as that of Figure 8. Also, the objective function value is not as good as the one of Figure 9 and somewhat inferior to that of Figure 8, meaning that the neighborhood mutation tends to produce compact results. The result of Figure 10 was obtained within 650 generations.

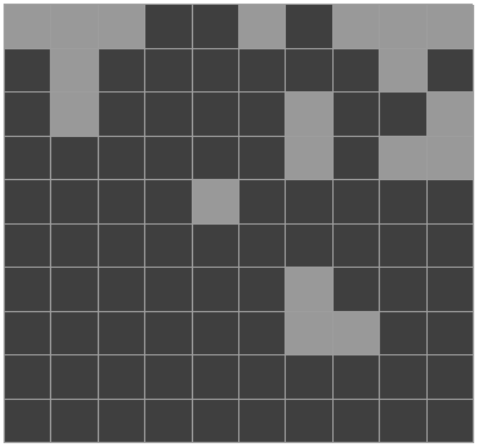


Fig. 10. Natural GA with NM

The above considerations motivate a combined implementation of NR and NM within the setting of the natural GA. The result is shown in Figure 11, in which the resulting arrangement is clearly compact but not connected as the one in Figure 8. Its objective function value is better than the one of Figure 9 and close, although inferior to that of Figure 10.
Finally, the operative GA was combined with the NR. As explained in section 3.2, the operative algorithm produces a new configuration after every generation. This configuration was each time subjected to the NR operator and an improved arrangement resulted. The final result is shown in Figure 12 and it is the same as that of Figure 8. The difference lies in the fact that the result is now obtained within 60 instead of 160 generations.

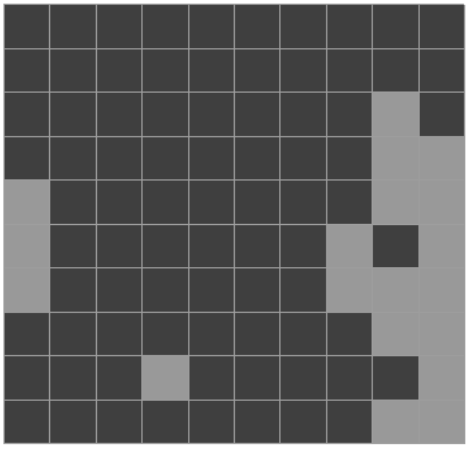


Fig. 11. Natural GA with NR and NM

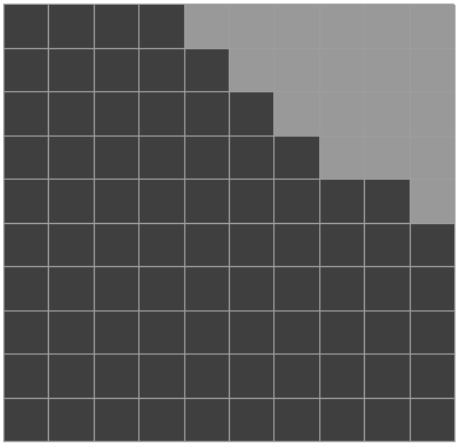


Fig. 12. Operative genetic algorithm with NR

6. Discussion – Conclusions

Spatial optimization problems are computationally intensive and demand the application of heuristic search methods for their solution. However, as it is demonstrated here, these methods have to be designed in accordance with the spatial character of the field under study. This character is fittingly modeled by means of cellular automata.

On a cellular background it is easier to pursue a balance between local and global characteristics. In this chapter two basic approaches are presented along this line. One of them is a natural genetic algorithm and, as described above, its typical chromosome consists of genes corresponding to land blocks. This natural algorithm is further equipped with operators acting on the local level and improving drastically the efficiency of the algorithm. The other approach has been characterized as operative, because the chromosomes of the genetic algorithm act as operators on the current configuration.

The latter method is in full accordance with the cellular automaton model, because the chromosome - operators act on the various cells in a way related to the neighbors of the cell. It is demonstrated that the operative genetic algorithm is more prone to producing compact configurations, although not necessarily yielding globally optimal results. On the other hand, these compact arrangements are highly desirable results in spatial optimization applications.

In summary, the schemes presented here provide the spatial planner with tools for obtaining a variety of alternative solutions with enhanced compactness characteristics without imposing special constraints. Indeed, these desirable results come out in an emergent sense. This means that no explicit provision is made within the computational process for their attainment. Such a provision would be the augmentation of the objective function by suitable penalty terms or the addition of specific constraints in the formulation of the problem. Moreover, the local operators (NR and NM) do not, at least explicitly, contain by their definition any bias or direction toward producing compact or contiguous arrangements. Locality of operation is their only characteristic.

Thus, it appears that methods connected to the cellular nature of the problem produce results more agreeable to a real-world point of view of spatial planning. This fact motivates further research toward a more systematic comparison of the present unconstrained methods to those involving explicit compactness, contiguity or percentage constraints. Clearly, the issue of emergence in the present algorithms calls for more investigation.

On the other hand, there can be constraints in relation to the hydraulic aquifer problem involved in the resource distribution question. Such constraints may answer the demand for ecological considerations. An example of this kind of constraint is the imposition of upper bounds on the pumping drawdowns at selected places of the aquifer. This restriction is aimed at protecting the aquifer from depletion.

Another direction of research concerns the multi-objective versions of the methods presented here. The operative genetic algorithm, as well as the local operators, admit non-trivial extensions into the multi-objective optimization field.

The progress of the cellular automaton toward optimal configurations raises the issue of self-organizing evolutionary methods and the role of an accompanying genetic algorithm in a mixed cellular – genetic scheme. This can be the subject of further follow-on research.

The integration of simulated annealing into the methods presented here is also worth investigating. A first step has already been taken in that direction (Sidiropoulos & Fotakis, 2009).

Finally, the present methodological framework may be applied to a variety of composite spatial planning problems arising in forest management, plant location and many other environmental or industrial fields.

7. References

Adamatzky, A.I., 1996. "Computation of shortest path in cellular automata", Math. Comput. Modelling, 23(4), pp. 105-113.

Aerts, J., Van Herwijngen, M., Janssen, R. and Stewart, T., 2005. Evaluating Spatial Design Techniques for Solving Land-Use Allocation Problems. Journal of Environmental Planning and Management, 48(1), pp.121-142.

Aerts, J.C.J.H. and Heuvelink, G.B.M. Using Simulated Annealing for Resource Allocation. International Journal of Geographical Information Science, 16(6), pp. 571-587, 2002.

Afshar, M.H. and Shahidi, M., 2009. "Optimal solution of large-scale reservoir-operation problems: Cellular-automata versus heuristic-search methods", Engineering Optimization,41(3), pp. 275-293.

Balmann, A., 1997. "Farm-based Modelling of Regional Structural Change: A Cellular Automata Approach", European Review Agricul. Econom. 24, pp. 85-108.

Colasanti, R.I. and Grime, J.P., 1993. "Resource Dynamics and Vegetation Processes: A Deterministic Model Using Two-Dimensional Cellular Automata", Functional Ecology 7, pp. 169-176.

Datta, D., Deb, K., Fonseca, C.M., Lobo, F.G., Condado, P.A. and Seixas, J., 2007. Multi-Objective Evolutionary Algorithm for Land Use Management Problem. International Journal of Computational Intelligence Research, 3(4), pp.371-384.

Deadman, P., Brown, D.R. and Gimblett, H.R., 1993. "Modelling Rural Residential Patterns with Cellular Automata", J. Environ. Management, 37, pp. 147-160.

Fotakis, D., 2009. "Spatial planning for decision making. Integrated land use and water resources management using evolutionary methods" Doctoral Dissertation, Aristotle University

Gaylord R. J. and Nishidate, K., 1996. *Modeling Nature. Cellular automata simulations with Mathematica.* Springer, 1996

Goldberg, D.E., 1989. *Genetic algorithms in Search, Optimization and Machine Learning,* Addison-Wesley.

Green, D.G., House A.P. and House S.M., 1985. "Simulating Spatial Patterns in Forest Ecosystems", Math. Comput. Simulation, 27, pp. 191-198.

Heinonen, T. and Pukkala, T., 2007. The use of cellular automaton in forest planning. Canadian Journal of Forest Research, 37, pp. 2188-2220

Hinton, G. E. & Nowlan, S., 1987."How learning can quide evolution", Complex Systems, 1, pp. 495-502.

Hogeweg, P., 1988. "Cellular Automata as a Paradigm for Ecological Modeling", Applied Mathematical Computations, 17, pp.81-100.

Jennerette G. D and Wu, J., 2001. "Analysis and simulation of land-use change in the central Arizona-Phoenix region, USA", *Landscpe Ecology*, 16, pp.611-626.

Karafyllidis I., and Thanailakis A., 1997. "A model for predicting forest fire spreading using cellular automata", Ecological Modelling, 99, pp. 87-97.

Karafyllidis, I. ,2004. "Design of a dedicated parallel processor for the prediction of forest fire spreading using cellular automata and genetic algorithms", *Engineering Applications of Artificial Intelligence*, 17, pp. 19-36.

Khan S., O' Connel N., Rana T. and Xevi E., 2008. "Hydrologic-economic model for managing irrigation intensity in irrigation areas under watertable and soil salinity targets", Environmental Modeling & Assessment, 13, pp. 115-120.

Krznowski,R. & Raper, J., 2001. *Spatial evolutionary modelling*, Oxford University Press.

Ligmann-Zielinska A., Church, R.L. and Jankowski, P., 2008. Spatial optimization as a generative technique for sustainable multiobjective land use allocation. International Journal of Geographical Information Science, 22(6), pp.601-622

Magalhaes-Mendes J., 2008. "Project scheduling under multiple resources constraints using a genetic algorithm", WSEAS Transactions on Business and Economics, 5(11), pp. 487-496.

Mathey, A.E., Krcmar, E., Dragicevic, S. and Vertinsky, I., 2008. "An object-oriented cellular automata model for forest planning problems", Ecological Modelling, 212, pp. 359-371.

Michalewicz, Z., 1992. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer.

Mitchel, M., Crutchfield, J. P. and. Hraber, P. T., 1994. "Evolving cellular automata to perform computations", *Physica,* D75:361-391

Moon, B. R., Lee, Y.S. and. Kim, C.K. , 1997. "Genetic VLSI circuit partitioning with two-dimensional geographic crossover and zigzag mapping", *Proceedings of the 1997 ACM symposium on Applied computing,* San Jose, California, United States, pp 274-278, 1997

Ortega J. F., Valero de Juan J.A., Tarjuelo J. M. and Lopez E. M., 2004. "MOPECO: an economic optimization model for irrigation water management" , Irrigation Science, 23, pp. 61-75.

Prasad, K., 1988. "Parallel Distributed Processing Models for Economic Systems and Games", Comput. Sci. Econom. Management, 1, pp. 163-174.

Sadeghi, S.H.R., Jalili, K. and Nikkami, D., 2009. Land use optimization in watershed scale. Land Use Policy, 26, pp.186-193.

Seppelt, R. and Voinov, 2002. A., Optimization methodology for land use patterns using spatially explicit landscape models. Ecological Modelling, 151, pp. 125-142

Sidiropoulos, E. and Fotakis, D., 2009. Cell-based genetic algorithm and simulated annealing for spatial groundwater allocation. WSEAS Transactions on Environment and Development, vol. 5 (4), pp. 351-360.

Sidiropoulos, E. and Tolikas, P., 2008. "Genetic algorithms and cellular automata in aquifer management", *Applied Mathematical Modelling*, 32(4), pp. 617-640.

Sole R.V. and Manrubia C.S., 1995. "Are Rainforests Self-Organized in a Critical State?", J. Theoret. Biology, 173, pp. 31-40.

Strange, N., Meilby, H. and Bogetoft, P., 2001. Land use optimization using self-organizing algorithms. Natural Resource Modeling, 14(4), pp.541-573

Strange, N., Meilby, H. and Thorsen, J.T., 2002. "Optimizing land use in afforestation areas using evolutionary self-organization", Forest Science, 48(3), pp. 543-555.

Supratid S. and Sadananda, R., 2004. "Cellular Automata - Critical Densities on Forest Fire Dispersion", *WSEAS Transactions on Computers*, Issue 6, vol. 3.

Vanegas, P. Cattrysse, D. Van Orshoven, J., 2010. Compactness in Spatial Decision Support: A Literature Review. Lecture Notes in Computer Science 6016, pp.414-429, Springer.

Wolfram, S. 2002. *A new kind of Science*, Wolfram Media Inc.

Yeo, I.-Y. and Guldmann, J.-M., 2010. Global spatial optimization with hydrological systems simulation: application to land use allocation and peak runoff minimization. Hydrology and Earth Systems Science, 14, pp. 325-338.

Yeo, I.-Y., Gordon, S.I. and Guldmann, J.-M., 2004. Optimizing patterns of land use to reduce peak runoff flow and nonpoint source solution with an integrated hydrological and land use model. Earth Interactions 8, paper no 6, pp.1-20.

Zhou H., Peng H. and Zhang C., 2007. "An interactive fuzzy multi-objective optimization approach for crop planning and water resources allocation", Lecture Notes in Computer Science, 4688, pp. 335-346.

5

CA City: Simulating Urban Growth through the Application of Cellular Automata

Alison Heppenstall[1], Linda See[1,2], Khalid Al-Ahmadi[3] and Bokhwan Kim[4]
[1]*University of Leeds,*
[2]*International Institute of Applied Systems Analysis (IIASA),*
[3]*King Abdulaziz City for Science and Technology (KACST),*
[4]*Urban Culture and Landscape Division, Ministry of Land, Transport and Maritime Affairs,*
[1]*UK*
[2]*Austria*
[3]*Saudi Arabia*
[4]*South Korea*

1. Introduction

It is estimated that 3.5 billion people currently live in cities, which equates to more than 50% of the total population (UN, 2010). These cities vary in size from 500,000 to over 10 million inhabitants (termed mega-cities). The number of cities, in particularly mega-cities, are set to significantly increase by 2050. However, this rapid urban population growth is taking place at differing speeds and spatial scales across the globe. Developing countries are experiencing a much higher level of urbanisation than developed countries. For example, by 2050, it is expected that Europe will be 84% urbanised (compared to 82% in 2010), whilst Asia and Africa will be 65% and 62% urbanised by 2050, respectively, compared to 40% in 2010 (UN, 2010).

Such rapid urbanisation, normally unplanned and spontaneous, brings its own set of problems, in both the social and physical environment. These include spatial segregation of the rich and poor, shortages in urban housing and basic services, and the production of vast volumes of waste and harmful synthetic materials (Pacione, 2005), as well as urban poverty. Psychologically, urbanisation can engender feelings of loneliness, self-centeredness, loss of a sense of community, and increasing crime rates (Knox, 1994).

To mitigate for these types of problems as well as to manage and plan future urban growth, a range of modelling techniques has been applied. The development of urban theory and modelling has a long history, e.g. *Industrial Location Theory (*Weber, 1909), *Central Place Theory (*Christaller, 1933), *the Concentric Zone Model (*Burgess, 1925*), the Sector Model (*Hoyt, 1939*)* and the *Multiple Nuclei Model (*Harris and Ullman, 1945*).* These classical theories and models have formed the foundation for studying urban structure and growth, but they have been criticised for being overly simplistic, unrealistic in their assumptions, and not applicable to the structure of today's cities (Chapin and Kaiser, 1979; Briassoulis, 2000; Batty, 1994, 1996). Another major criticism levelled at these models is their static nature. They are unable to explain the spontaneous growth that has taken place in modern cities under a non-equilibrium status, which has evolved diversely from highly dispersed edge cities to

huge mega-cities (Batty, 1994). This last point reflects how, with a deeper understanding of urban phenomena, scientists have begun to recognise that cities are not uniform or a single type of phenomenon. Instead they are increasingly being recognised as complex systems through which non-linear processes, emergence and self-organisation occur (Allen 1997; Portugali 2000; Batty 2007). These processes shape the spatial growth of the city over time with structured and ordered patterns emerging (Torrens 2000a).

More recent urban models, catalysed by the availability of mainframe and desktop computers, have focused on modelling the spatial structure of urban growth processes and the notion of self-organisation. These models include large-scale urban models of land use-transportation developed using a range of analytical methods from other disciplines including linear programming, physics, human ecology, mathematics, operations research, regional science and economics (Batty, 1981; Klosterman, 1999; Torrens, 2000b). Many of these models have incorporated cellular automata (CA), whether to model land parcels or the actions of firms or organisations.

This chapter will demonstrate the potential of CA as a tool for urban planning and development using two CA models and case studies, one from Saudi Arabia and the other from the Republic of Korea. In Riyadh, Saudi Arabia, the government are planning new suburban towns to absorb future population growth. The exact location of these towns is subject to several environmental constraints. CA have been used as a planning tool to evaluate several sites and visualise the likely future growth of these towns. The impact of increased population growth is also one of the main drivers for the construction of a new city in the Republic of Korea. This chapter illustrates how CA have been used for the evaluation of different potential sites and to visualise how the city could grow over time based on various scenarios. A review of previous work in this area is first provided to place these modelling exercises in context. This is followed by a description of the case studies, the results of the model scenarios and a comparison of the two models. The strengths and weaknesses of the models are discussed including areas for further development.

2. Previous research

Cellular Automata (CA) provide a way of simulating complex systems and self-organising processes over space and time (Wolfram, 1994). As a result of their capability for generating complex patterns through local rules, and for linking rules to their consequences, CA can provide insights into the different pathways that control and form systems. The use of CA for modelling urban dynamics and growth has been the subject of considerable research over the last two decades. A comprehensive review has recently been undertaken by Santé et al. (2010), which provides a historical overview of the use of CA in urban modelling, covering CA as a largely theoretical approach to urban simulation (e.g. Itami, 1998; Batty, 1998) as well as early attempts at applying CA models to real cities (e.g. Engelen et al., 1995; White et al., 1997). Since then many different examples have emerged, which have been built using different structures and developed for diverse applications (e.g. see the work of Li et al. (2003), Barredo et al. (2004) and Stevens et al. (2007) to name a few).

In their review, Santé et al. (2010) examined thirty-three urban CA models from the literature. To compare these different models, nine main characteristics were chosen including the purpose of the model, the resolution, the predicted states, the neighbourhood, the transition rules, the constraints, the methods of calibration and validation, and any integration with other models. In addition, they also analysed the factors or drivers of urban

growth employed by each of the models. Based on these analyses, they were able to discuss some of the main strengths and weaknesses of the use of CA for urban modelling. The authors argue that the simplicity of CA models is considered both a strength and a weakness. It is difficult to capture the complexity of urban systems within such a simple representation, and for this reason, many modifications or relaxations have been made, which brings into question whether many of the models are still actually CA. The model flexibility in terms of adaptation to various real world situations is also discussed so models which have rules and parameters calibrated through data mining methods are less flexible than more generically specified transition rules. Less than 40% of the models predicted multiple land uses while most simply determined whether cells are urbanised or not. In terms of model accuracy, they found that overall this was generally very good, but one area where further research is needed is in the development of new validation methods, especially in the area of pattern recognition. Other areas where the authors suggest further research includes more integration of CA modelling with urban and spatial theory as well as integrating different modelling types such as agent-based models in a hybrid representation. Since the period of the review (which covers research published up to early 2009), a number of new papers have appeared in the literature. However, much of this research has involved the application of existing CA urban models to different areas, or modifications to improve the model performance. For example, the SLEUTH model (Clarke et al., 1997) continues to be applied to different parts of the world. Rafiee et al. (2009) used the SLEUTH model for simulating urban growth in Mashad City, Iran, Jantz et al. (2010) have made improvements to SLEUTH in the development of a fine scale regional model of the Chesapeake Bay area in the eastern US, while Wu (2009) applied the SLEUTH model to the Shenyang metropolitan area of China. Similarly, the CLUE-S model (Veldkamp and Fresco, 1996) has been used by Pan et al. (2010) and Zhang et al. (2010) for modelling areas in China with particular emphasis on the effects of scale on model outcomes, and the consideration of uncertainty. Other existing CA models used in recent work includes the research by Petrov et al. (2009), who applied the MOLAND model (Lavelle et al., 2004) to scenarios of future urban land use change in 2020 in the Algarve, Portugal, and Poelmans and Van Rompaey (2009), who applied the Geomod model (Pontius, 2001) to examine urban sprawl in the Flanders-Brussels region. Other areas of research have involved modifications to the basic CA structure, e.g. the use of a variable grid CA (van Vliet et al., 2009), a vector-based CA (Moreno et al., 2009) and hybrid model variants, e.g. Han et al. (2009), who integrated a systems dynamics model with a CA model, and Wu et al. (2010), who coupled neural networks with CA. There is also a trend towards the development of agent-based urban growth models, either in conjunction with CA (e.g. Wu and Silva, 2009) or as a new framework for modelling urban spatial dynamics (e.g. Irwin et al., 2009). Finally, recent work by Poelmans and Van Rompaey (2010) involved the comparison of a CA model against other approaches to modelling urban expansion including logistic regression and a hybrid approach that combined both individual approaches. When considering the results at only one resolution, the hybrid approach produced the best result. However, when multiple resolutions were considered, the logisitic regression proved to be superior. This work once again emphasises the importance of scale, which has been considered in more recent work as outlined above.

Thus, it is clear from the growing literature that the use of CA will continue to be used for modelling urban growth, whether building upon existing models or in hybrid formulations.

3. CA model of urban growth in Riyadh, Saudi Arabia

The first case study will assess the likely impact of several new towns and satellite centres around the city of Riyadh, Saudi Arabia. Due to the discovery of oil, Riyadh has experienced significant growth over the last 60 years. The population of Riyadh has increased from 25,000 in the 1930s to 2.5 million by the early 1990s. The current rate of population growth is around 8.1% per annum and the city is expected to reach 10 million people by 2020. The spatial expansion of the city has also seen dramatic changes, growing from a geographical extent of less than 1 km² in the 1920s to over 1,150 km² by 2004 (ADA, 2004).
To manage this high rate of urban growth and change, the Saudi Arabian government has instigated a series of master plans for Riyadh. The main aim of these plans is to re-structure and direct urban expansion to achieve sustainable development in the future (ADA, 2004). To ensure that the master plan will have beneficial effects on the urban fabric and population of the city, policy makers require a planning support tool that has the capability of simulating the complexities of managed urban growth over the next 15 years. The model outlined by Al-Ahmadi et al. (2009a,c) has been developed with this requirement in mind.

3.1 CA model details

The model, referred to as the Fuzzy Cellular Automata Urban Growth Model (FCAUGM), is a stochastically constrained CA. The model creates a new urban cell *ij* at time *t*+1 if the cell's development possibility (DP) score is greater than or equal to a transition threshold parameter, λ, as follows:

$$\text{If } DP_{ij}^{t} \geq \lambda \text{ Then } S_{ij}^{t+1} = \text{Urban, Otherwise} = \text{Non-Urban} \quad (1)$$

where S_{ij}^{t} is the state of a cell *ij* a time *t+1*; Dp_{ij}^{t} is the development possibility of a cell *ij* a time *t*; and λ is the transition threshold between 0 and 1, which is determined through calibration.
The model defines the state of a cell S_{ij}^{t+1} at *t+1* as a function of its development possibility (*DP*) at time *t*, which is a function of both the development suitability (*DS*) of a cell *ij* at time *t*, and a stochastic disturbance factor (*SDF*):

$$S_{ij}^{t+1} = f\left(DP_{ij}^{t}\right) = (DS_{ij}^{t} * SDF) = (DS_{ij}^{t} * [1 + \ln(\gamma)^{\alpha}]) \quad (2)$$

where DS_{ij}^{t} is the development suitability of a cell *ij* a time *t*; γ is a uniform random variable within the range [0,1]; and a is the dispersion parameter which controls the size of the stochastic perturbation. The development suitability, DS_{ij}^{t}, of a cell *ij* is a function of four driving forces which potentially contribute and affect the spatial patterns of urban growth:

$$DS_{ij}^{t} = f\left(TSF_{ij}^{t},\ UAAF_{ij}^{t},\ TCF_{ij}^{t},\ PPRF_{ij}^{t}\right) \quad (3)$$

where TSF_{ij}^{t} is the transport support factor of a cell *ij* at time *t*; $UAAF_{ij}^{t}$ is the urban agglomeration and attractiveness factor; TCF_{ij}^{t} is the topographical constraint factor; and $PPRF_{ij}^{t}$ is the planning policies and regulation factor. The four driving forces of urban growth (TSF, UAAF, TCF and PPRF) are themselves functions of fuzzy input variables as shown below:

$$TSF_{ij}^{t} = f\left(ALR_{ij}^{t},\ AMR_{ij}^{t},\ AMJR_{ij}^{t}\right) \quad (4)$$

$$UAAF_{ij}^{t} = f\left(UD_{ij}^{t},\ AECSES_{ij}^{t},\ ATC_{ij}^{t}\right) \tag{5}$$

$$TCF_{ij}^{t} = f\left(G_{ij}^{t},\ A_{ij}^{t}\right) \tag{6}$$

$$PPRF_{ij}^{t} = f\left(PA_{ij}^{t}, EA_{ij}^{t}\right) \tag{7}$$

where TSF is determined by Accessibility to Local Roads (ALR), Accessibility to Main Roads (AMR) and Accessibility to Major Roads (AMJR); the UAAF is a function of Urban Density (UD), Accessibility to Town Centres (ATC) and Accessibility to Employment Centres and Socio-Economic Services (AECSES); the TCF is determined by Gradient (G) and Altitude (A); while the PPRF indicates the Planned Areas (PA) and Excluded Areas (EA). The drivers of urban growth are linked together via a set of fuzzy rules, which are determined during calibration along with the membership functions. A fuzzy inference engine combines the fuzzy rules to create a fuzzy development suitability score (DS_{ij}^{t}), which is then defuzzifed to a crisp value as shown below:

$$DS_{ij}^{t} = 1 + \frac{\sum_{i=1}^{n} \mu c_n(z_{ij}) * z_{ij}}{\sum_{i=1}^{n} \mu c_n(z_{ij})} \tag{8}$$

where μc_n is the membership function of the development suitability of a rule n and z_{ij} is the fuzzy output value (development suitability) of a cell ij of a rule n.

The model was then calibrated using a sample dataset from the study area, chosen using a disproportional stratified random sampling method consisting of 60% urban and 40% non-urban locations. A genetic algorithm and a parallel implementation of simulated annealing were used as optimisation methods. In addition, experts were used to determine the membership functions and rules as a third method for comparison. The mean squared error and the root mean squared error were used as objective functions in the calibration. They were combined into a single, weighted and standardised objective function to penalise situations where model parameters fall outside the allowable bounds. The genetic algorithm provided the best calibration results.

Once calibrated, the FCAUGM was used to simulate urban growth in Riyadh city for the following three periods: 1987–1997, 1997–2005 and 1987–2005 in order to carry out validation. Several quantitative measures to determine model performance were used such as overall accuracy (based on a confusion matrix), the Lee-Sallee index and a spatial pattern measure. In terms of overall accuracy, the FCAUGM performed well, i.e. 93% for 1987–1997, 92% for 1987–2005 and 94% for the combined period 1987-2005. However, when considering only urban agreement, the accuracy drops to 52.5% in 1987-1997, 37.6% in 1997–2005 and 74.3% in 1987–2005. For further details of the model including the full calibration and validation procedures, the reader is referred to Al-Ahmadi et al. (2009a,c).

3.2 Outline of scenarios

Two scenarios using the FCAUGM are presented in this chapter: (i) the development of new satellite towns, and (ii) the creation of new metropolitan sub-centres. These different planning scenarios form part of current Saudi government planning policy, referred to as the Metropolitan Development Strategy for Arriyadh (MEDSTAR). According to the HCDR (2004), two locations within Riyadh's urban boundary have been allocated for development as new towns. The northern town will accommodate almost 1 million people with the eastern town absorbing 900,000. The position of the towns has been selected to maximise

both business opportunities (proximity to airports and major infrastructure) and development of high quality residential areas.
Another aim of the MEDSTAR strategy is to decrease the concentration of the city's activities and services from one place to a more decentralised approach. This strategy has been translated into the designation of five metropolitan sub-centres located 25 km from the city centre. The goal of these new centres is sustainable urban development for the Riyadh metropolitan area through restructuring and urban growth. This is to be achieved through the creation of urban hubs as the focus for employment, investment, commercial facilities and governmental administrative functions.

4. CA model of new city growth in the Republic of Korea

The second case study is the development of an urban growth model for a new city in the Republic of Korea. In 2003, the Korean government announced plans to construct a new administrative capital near Daejun city. This new city would alleviate the problems that are currently caused by economic concentration and population agglomeration in the existing capital and surrounding area. The city, due for completion in 2014, will cost the Korean government approximately £4.2 billion. On completion of the city, 12 out of 18 ministries will move to the new administrative capital city. The population is projected to grow to 500,000 by 2030. To date, this is the single most important urban development ever planned in the Republic of Korea. However, there has been little modelling undertaken to determine how the city will develop or spread. Previous new city development has resulted in the spread of urban land use into conservation zones and unplanned regions. A model was therefore designed for planners to experiment with different growth scenarios to avoid these problems.

4.1 CA model details

The CA model used in this case study is called the NCGM (New City Growth Model). As with the FCAUGM developed for Riyadh, it is also a type of stochastically constrained CA urban model, operating as outlined in equation 1. However, there are a number of differences. The first is in the form of the stochastic disturbance factor, which uses a hybrid transformation function. Moreover, the input factors that drive urban growth in the NCGM have specifically been chosen for modelling new city growth and not growth due to natural processes. These factors include: urban density, road accessibilty, subway accessibility, slope, planning areas and excluded areas. The developmental suitability, referred to in this model as the composite score for a cell ij, $ComS_{ij}$, is a weighted summation of the factors as shown below:

$$ComS_{ij} = \left[\left(W_U * F_{ijU}\right) + \left(W_R * F_{ijR}\right) + \left(W_S * F_{ijS}\right) + \left(W_{Sub} * F_{ijSub}\right) + \left(W_P * F_{ijP}\right)\right] * F_{ijEx} \quad (9)$$

where $F_{ijU}, F_{ijU}, F_{ijU}, F_{ijSub}$ are the factor scores of the inputs urban density, road accessibility, slope and subway accessibility of ij respectively, F_{ijP} and $F_{ijEx}$are binary factors which state whether a cell is inside or outside of the planning area or the excluded areas, respectively, and $W_U, W_R, W_S, W_{Sub}, W_P$ are the weights of the inputs factors for urban density, road accessibility, slope, subway accessibility and planning areas. The composite score is then turned into a probability for development through the hybrid transformation function. The transition threshold then turns the probability into a developed or non-developed cell.

Similar to the FCAUGM, a genetic algorithm was used to determine the model parameters during calibration using the mean squared error as the objective function. In the case of the NCGM, the weights in equation 9 and the size of the neighbourhood were optimised. The data for calibration were taken from the new city of BanDung and were divided into three time periods: the Pre-Planning, the Mid-Planning, and the Post-Planning periods, respectively. Once the model was calibrated, data from the new city of IlSang were used for validation. Quantitative measures of model performance were similar to those used in the FCAUGM. The overall accuracies for the Mid-Planning and Post-Planning periods were 93% and 88%, respectively, although model performance decreased to 54% and 45%, respectively, when considering only urban areas. For further details of the structure of the model, and the calibration and validation procedures, the reader is referred to Kim (2005).

4.2 Outline of scenarios for new city growth

The first scenario, termed the baseline, simulates new growth from 2004 to 2020, which would take place without any explicit policy interventions. A default urban growth rate of 7.5% was used for both the Mid-Planning and Post-Planning periods. Due to data availability, the Mid-Planning period begins in 2004 and ends in 2012; this is according to the building plan as set out by the Ministry of Construction and Transportation (MOCT, 2005). The Post-Planning period covers the 8 years beyond the Mid-Planning period (from 2012 to 2020). A second scenario is then presented with an urban growth rate of 10%. The aim of this scenario is to estimate the potential areas to be developed if further urban growth, beyond that expected, takes place.

5. Model results

5.1 Modeling new development outside of Riyadh, Saudi Arabia

In the first scenario, the FCAUGM was run to examine the impact of two new satellite towns near Riyadh. The results of the simulation are shown in Figure 1. Here, the two proposed towns can be seen to absorb most of the urban development that is anticipated to take place to the southwest and south of Riyadh. It is highly likely that the new towns will attract residents with a preference for a calmer and more suburban living environment than is currently provided in the city. This suggests that development of the new satellite towns will be a positive planning intervention for the future of the city.

Assessing the impact of the new planned metropolitan sub-centres assumes that the development is dominated by the process of decentralisation and densification of activities and services in the different sectors of the city and particularly nearby sub-centres, i.e. the city is allowed to grow faster near sub-centres than the major urban area and town centre. Figure 2 shows that the development is more compact and concentrated around the sub-centres, resulting in more sustainable development. Thus, this scenario shows how unnecessary extended urban expansion or urban sprawl is prevented. It can be seen from the predicted urban form of Riyadh city under this scenario that the policy of designating five metropolitan sub-centres in the outer sectors of the city has to a large extent achieved the ultimate goals set by ArRiyadh Development Authority (ADA, 2004), i.e. re-structuring and direct urban expansion to achieve sustainable development in the future.

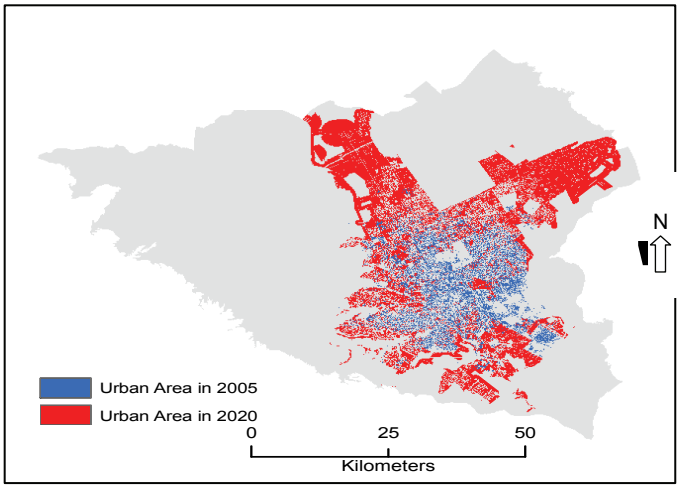


Fig. 1. Predicted urban growth of Riyadh under the New Satellite Cities Scenario

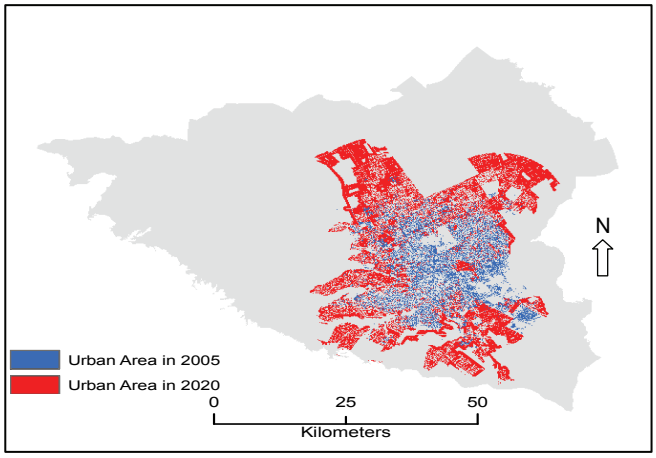


Fig. 2. Predicted urban growth of Riyadh under the New Metropolitan Sub-Centres Scenario

5.2 Development of a new city in the Republic of Korea

The results for the baseline scenario are shown in Figure 3. Figure 3a and Figure 3b are the final predicted images during the Mid-Planning and Post-Planning periods, respectively. The prediction during the Mid-Planning period started from the areas coloured in blue in Figure 3a, and the new urban areas are coloured in red in the same image. The results from Figure 3a were used as the initialisation/start point for the simulation of the Post-Planning period shown in Figure 3b. Figure 3c is an overlaid image combining urban growth during the Mid-Planning period (coloured in purple) and Post-Planning period (coloured in red). Finally, Figure 3d shows the predicted urban shape of the new administrative capital area in 2020.

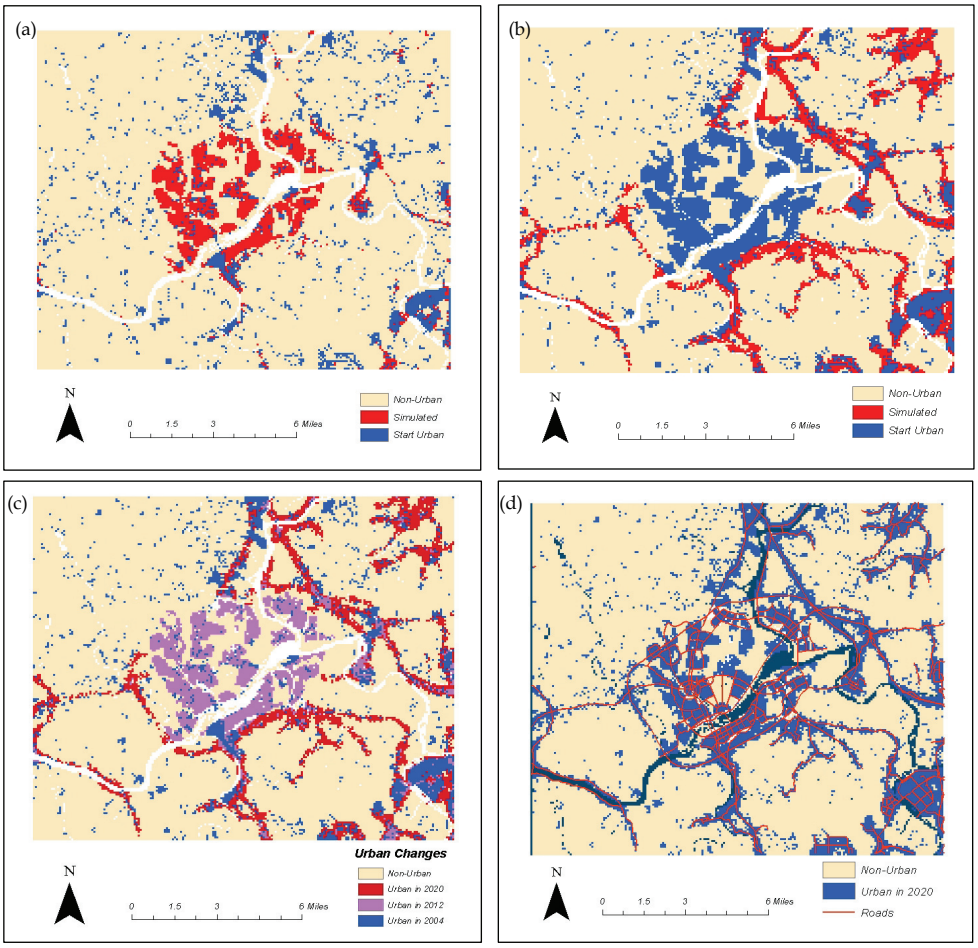


Fig. 3. Predicted urban growth of the new administrative capital area under the first (baseline) scenario during (a) the Mid-Planning period; (b) the Post-Planning period; (c) urban changes during both periods; and (d) the final urban shape in 2020.

During the Mid-Planning period, most of the urban development occurred within the area planned for the new administrative capital city with small amounts of urban clustering developing in the east and south-eastern areas (see Figure 3a). This might be considered the result of well organised and intended urban growth. However, during the Post-Planning period, further urban development would be expected to take place around the planned city, particularly along the road network (including the outer ring road and north-east corner of the image). Development pressure from the already developed area seems to cause the urban development in the south-eastern corner of Figure 3b. It is also of note that a new urban cluster has developed in the north-eastern corner of the image.

Figure 3c clearly demonstrates the differences between those urban growth patterns during the Mid-Planning and Post-Planning periods. Most urban development during the Mid-

Planning period (coloured in purple) is concentrated on the planned city. However, further urban development during the Post-Planning period would create rather sporadic forms of urban settlements along the road network (coloured in red). The predicted final urban shape in 2020 shows evidence of sporadic urban growth (see Figure 3d).

The second scenario is shown in Figure 4, where the aim is to examine the affect of a higher than expected urban growth rate of 10%. Figure 4a shows the predicted urban pattern for the Mid-Planning period. In comparison to the baseline scenario (Figure 3a), four types of urban growth can be detected. Firstly, urban development along the road crossing from the north to east edges of the image is apparent. Secondly, an urban cluster in the areas immediately to the south and south-east of the new administrative capital city develops considerably. Thirdly, two small urban clusters have grown in the north-east corner. Finally, a new urban development can be detected in the south-eastern corner of the image.

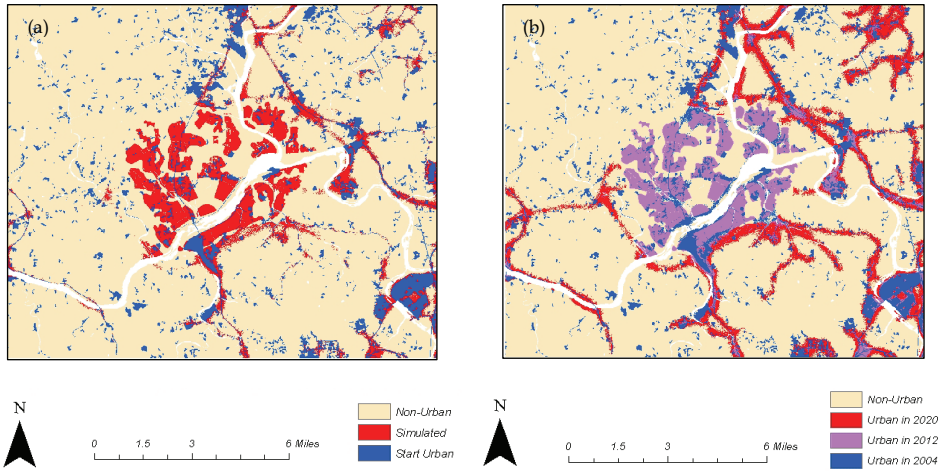


Fig. 4. Model simulation results using an urban growth scenario of 10% for a) the Mid-Planning period and b) the Post-planning period

Figure 4b shows the higher simulated level of urban growth during the Post-Planning period. Here, the result of the baseline scenario (7.5% urban growth) is coloured in purple for comparative purposes. The trends seen in the Mid-Planning period simulation are evident in this scenario. The two small urban clusters in the north-east corner that developed during the Mid-Planning period are clearly detectable, forming a large urban cluster. Away from that urban cluster, most urban development takes place along the road network linking the new administrative capital city to the other developed areas. Urban growth along the road crossing from the north to east edges of the image along with urban growth that happened in the areas immediately to the south and south-east of the new administrative capital city are now even clearer. The sporadic urban growth patterns along the road network found with the default urban growth rates (Figure 3a) is reinforced with application of a higher urban growth rate.

5. Comparison of the models

Both the FCAUGM and the NCGM have been developed independently at the University of Leeds to model urban growth for two very different areas. A comparison of the FCAUGM and NCGM is presented in Table 1 using the same characteristics as employed by Santé et al. (2010).

Model Characteristics	**FCAUGM**	**NCGM**
Objective	Descriptive + Predictive	Predictive
Cell space	20 m square cells	30 m square cells
States	Urban, non-urban	Urban, non-urban
Neighbourhood	Size is determined during calibration	Size is determined during calibration
Transition rules	Used the same exponential form as Li and Yeh (2001)	Hybrid transition rule that acts as sigmoidal in certain ranges and exponential in others
Constraints	Annual growth rate for urban land	Annual growth rate for urban land
Other methods	Genetic algorithm, parallel simulated annealing and expert knowledge used in calibration	Genetic algorithm used in calibration
Drivers of growth	Accessibility to local, main and major roads, urban density, accessibility to Town Centres, accessibility to employment centers and socio-economic services (AECSES), slope, altitude, planned areas, excluded areas	Urban density, accessibility to roads, slope, accessibility to the subway, planning areas, excluded areas
Calibration	Fuzzy membership function of growth drivers, the fuzzy rules and the transition threshold are calibrated using a genetic algorithm, simulated annealing and expert knowledge in three separate instances of the model	Weights of growth drivers and neighbourhood size are calibrated using a genetic algorithm
Validation	Visual inspection of the spatial patterns; overall accuracy from a confusion matrix; accuracy of urban areas; Lee-Sallee index; spatial pattern measure	Overall accuracy from a confusion matrix; accuracy of urban areas; Lee-Sallee index

Table 1. Comparison of the two CA urban growth models

The FCAUGM was designed to be highly generic, and can therefore be applied to any planning scenario. This is in part due to the requirements of this model to be flexible and extensible, for example modelling not only the development and growth of new towns around the city of Riyadh, but also other developments such as metropolitan sub-centres and other types of scenarios as outlined in Al-Ahmadi et al. (2009b). Santé et al. (2010) would actually consider such a model to be the least flexible when compared to other CA models because the model is calibrated using a genetic algorithm for local conditions. However, this may also explain why the accuracy of the model is high when compared to other CA models and therefore represents a tradeoff in this type of modelling. The NCGM, on the other hand, was specifically designed for simulating the likely growth of a planned new city in the future. The model is actually much simpler than the FCAUGM, with less parameters to calibrate and is therefore the more flexible of the two.

Both the FCAUGM and NCGM are predictive models as per the types of objective outlined in Santé et al. (2010). However, the FCAUGM has the added advantage of also being a descriptive model because of the fuzzy nature of the model formulation. The fuzzy logic component is intended to replicate human decision-making as well as the uncertainty around many of the drivers used in the model. The resulting fuzzy rules and membership functions are transparent so the effect of the drivers can be determined by examining the rules and the configuration of the membership functions. For example, in Al-Ahmadi et al. (2009c), different rules fired more frequently depending on which combination of drivers was tried in the model, which could then be related to the type of urban growth happening in a particular period.

The cell space or resolution of the two models is similar, with a slighter higher resolution for the FCAUGM, and both used square-shaped grid cells. As with most other urban CA models in the literature, the two models do not predict multiple land use types, just whether a cell is urbanised or not.

Both the FCAUGM and the NCGM are stochastically constrained CA but the transition rules differ. The FCAUGM uses an exponential form while the a hybrid transition rule was created for the NCGM that overcomes problems associated with calibration and the use of the mean squared error as the objective function. The hybrid rule was simpler to calibrate and can also be interpreted in terms of the rationality of the decision maker. Both models use assumptions about the annual growth rate, whether this is based on the past or an increased rate of growth depending on the scenarios run. Both models have been calibrated using a genetic algorithm although the FCAUGM was also calibrated using a parallel implementation of simulated annealing and expert knowledge. Three instances of the FCAUGM were created and it was found that the genetic algorithm gave the best result.

Both models use some of the same drivers of growth (e.g. accessibility to roads, urban density, slope, planned and excluded areas) but they differ in others (e.g. accesibility to town centers and socio-economic services, accessibility to employment and accessibility to the subway). This is partly a reflection of the difference in purpose, i.e. the more generic model that is the FCAUGM compared to the model specifically designed to examine new growth, but also a reflection of individual areas, the strategies governing development and their development history to date.

Both models used similar measures of model performance in validating the model, once calibrated. Overall accuracy is one of the most commonly used methods, and both models performed very well, especially in relation to other examples cited in Santé et al. (2010). Both performed less well when taking only urbanised areas into account but still had acceptable

performance. However, overall accuracy is a single global measure and does not take the resulting spatial patterns into account. Both models therefore also used the Lee-Sallee index, which is one method of trying to capture the urban shape of the output; however, it only works on immediate neighbours, and therefore does not capture higher level clustering or structure. The FCAUGM was further validated using a spatial pattern measure that looks at agreement within a defined neighbourhood but it is clear that measures which capture the spatial structure in a more realistic way are still needed. Finally, there were no experiments with changing the scale of the simulation and therefore determining the effects at multiple resolutions.

6. Conclusions

Figures from the UN (2010) show that the increase in urban population over the next 40 years will be dramatic (from 3.4 billion in 2009 to an estimated 6.3 billion in 2050). The ability to forecast and understand the impact of this growth on cities will be a major directive of government planning legistation and policy making. How can the growth in exisiting cities be managed so as not to create social, economic and environmental disparaties for their inhabitants? Is it possible when designing a new city, as in the Korean case study, to use modelling tools to simulate likey growth under a variety of scenarios; could this pave the way for sustainable growth? The work presented within this chapter has taken two constrasting areas of urban growth, i.e. expansion of an exisiting urban area and creation of a new city, and presented results that support the use of CA as an urban modelling and simulation tool to provide answers to some of these questions.

In the Saudi Arabian example, the FCAUGM model, a CA driven model, was presented and the impact of the development of new satellite towns, and the creation of new metropolitan sub-centres around the city of Riyadh was assessed. Both these simulations are part of current Saudi government planning policy. A separate CA driven model called NCGM, was developed and applied to the planning of a new city in the Korean example. The results from both case studies demonstrate that the models are capable of predicting plausible patterns of future urban growth. Furthermore, both models could be adapted for use as a spatial planning support tool for urban planners and decision makers in both Saudi Arabia and the Republic of Korea. Such tools can assist in testing out plans, policies and other factors underpinning and influencing processes of urban growth. This can in turn lead to a better understanding of the factors influencing urban growth and ultimately allow the evaluation of the consequences of diverse future scenarios for urban growth by answering '*what if*' type questions (Yeh and Li, 2001).

The use of CA in both these examples reflects a shift in how cities are now viewed; they are not seen as static entities, but dynamic and complex systems composed of spatial and temporal interactions between the elements (people, environments, and policies, etc.). CA rooted in complexity theory, have shed light on modelling complex urban systems by allowing the components within the model to reflect spatial (neighbourhoods and constraints) and temporal (update or feedback) interactions. The FCUAGM and NCGM models are representative of a new breed of CA model that can be found in the literature. Both can be classed as hybrid models drawing on the strengths of other methodologies, for example genetic algorithms for calibration and fuzzy logic for formulation of the transition rules. Through this hybrid approach, these models can be significantly better at evaluating

the past, present and future consequences of urban planning interventions than their predecessors.

However, despite the advantages that these models bring, there are still areas that can be further improved. An obvious inclusion to both models would be an Agent-Based Model (ABM) for simulating populations. Agents are an increasingly familiar and powerful tool amongst geographers and social scientists. Within the context of these models, agents could be used to represent population dynamics at several different spatial scales such as individuals, households or neighbourhoods (it should be noted that ABMs are not just limited to representating population dynamics, they could be used for representating a varety of diverse entities such as firms, planners and governments). ABM not only compliments the bottom-up notion of modelling supported by CA, but would strengthen the realism of urban models by allowing greater detail to be included. However, this realism comes at a price. For this type of application, ABMs would require large volumes of accurate demographic and behavioural data. Often these data are not available, particularly for developing countries, and where it is available, the computational efficiency of the model (dependent on whether the model is used in academic or real world planning) should be considered as being of crucial importance.

There are other areas that to which future research should be directed. The effect of scale on these models, i.e. how to track the impacts of local neighbourhood processes on the patterns that emerge at city-wide level is an area that is ripe for investigation. How can self-organisation (a strong component of these models) be best visualised and understood? Are there better ways that these models can be validated and calibrated? Both models use genetic algorithms for calibration but are there better, more efficient optimisation methods available? How can pattern recognition techniques be used to improve validation?

Kirkby *et al.* (1992, p. 3) suggested that "models can never fully represent the real world, but can only be analogies or analogues which have some features and behaviours in common with it". The models presented in this chapter are not exceptions. Despite this caveat, both models were developed, calibrated, and tested to support different aspects of urban growth planning. Both models are driven by CA and have been shown to successfully simulate likely future growth. Inclusions of ABM and new calibration techniques can only improve the realism of these models and aid in the planning and management of sustainable future cities.

7. References

Al-Ahmadi, K., Heppenstall, A.J., Hogg, J. & See, L. (2009a). A Fuzzy Cellular Automata Urban Growth Model (FCAUGM) for the city of Riyadh, Saudi Arabia. Part 1: Model Structure and Validation. *Applied Spatial Analysis*, Vol.2, No.1, 65-83, ISSN 1874-463X.

Al-Ahmadi, K., Heppenstall, A.J., Hogg, J. & See, L. (2009b). A Fuzzy Cellular Automata Urban Growth Model (FCAUGM) for the city of Riyadh, Saudi Arabia. Part 2: Scenario Analysis. *Applied Spatial Analysis*, Vol.2, No.2, 85-105, ISSN 1874-463X.

Al-Ahmadi, K., See, L., Heppenstall, A. & Hogg, J. (2009c). Calibration of a fuzzy cellular automata model of urban dynamics in Saudi Arabia. *Ecological Complexity*, Vol.6, No.2, 80–101, ISSN 1476-945X.

Allen, P.M. (1997). *Cities and Regions as Self-organizing Systems: Models of Complexity*, Gordon and Breach Science, ISBN 3790819360, Amsterdam.

Arriyadh Development Authority (ADA). (2004) *Arriyadh Metropolitan Strategy Plan: Part 2 State of the City, Background and Issues*. Riyadh, Saudi Arabia.

Barredo, J.I., Demicheli, L., Lavalle, C., Kasanko, M. & McCormick, N. (2004). Modelling future urban scenarios in developing countries: an application case study in Lagos, Nigeria. *Environment and Planning B*, Vol.31, No.1, 65-84, ISSN 0265-8135.

Batty, M. (1981) Urban models, In: *Quantitative Geography: a British View,* Wrigley, N. & Bennett, R.J. (Eds.), 181-191, Routledge and Kegan Paul, ISBN 0-7100-0731-0, London.

Batty, M. (1994). A chronicle of scientific planning: The Anglo-American modeling experience. *Journal of American Planning Association*, Vol.60, No. 1, 7-16, ISSN 0194-4363.

Batty, M. (1996). Visualizing urban dynamics, In: *Spatial Analysis: Modelling in a GIS Environment*, Longley, P. & Batty, M., (Eds.), 297-320, Geoinformation International, ISBN , ISBN 0-470-23615-9, Cambridge.

Batty, M. (1998). Urban evolution on the desktop: simulation with the use of extended cellular automata. *Environment Planning A*, Vol.30, No.11, 1943-1967, ISSN 0308-518X.

Batty, M. (2007). *Model Cities*. Centre for Advanced Spatial Analysis, University College London, Working Paper 113,
http://www.casa.ucl.ac.uk/publications/workingPaperDetail.asp?ID=113.

Briassoulis, H. (2000). Analysis of land use change: theoretical and modeling approaches, In: *The Web Book of Regional Science* URL: http://www.rri.wvu.edu/regscweb.htm, Loveridge, S. (ed.), Regional Research Institute, West Virginia University, Morgantown, USA.

Burgess, E.W. (1925). The growth of city: an introduction to a research project, In: *The City*, Park, R.,E., Burgess, E.W. & McKenzie, R.D. (Eds.), 47-62, The University of Chicago Press, ISBN 0226646114, Chicago.

Chapin, F.S. & Kaiser, E.J. (1979). *Urban Land Use Planning*, University of Illinois Press, ISBN 0252021010, Urbana, IL.

Christaller, W. (1933) *Central Places in Southern Germany*, Englewood Cliffs, ISBN-10: 0131226304, New Jersey.

Clarke, K.C., Hoppen, S. & Gaydos, L. (1997). A self-modifying cellular automaton model of historical urbanization in the San Francisco Bay area. *Environment and Planning B,* Vol.24, No.2, 247-261, ISSN 0265-8135.

Engelen, G., White, R., Uljee, I. & Drazan, P. (1995). Using cellular automata for integrated modelling of socio-environmental systems. *Environmental Monitoring and Assessment*, Vol.34, No.2, 203-214, ISSN 0167-6369.

Han, J., Hayashia, Y., Caob, X. & Imura, H. (2009). Application of an integrated system dynamics and cellular automata model for urban growth assessment: A case study of Shanghai, China. *Landscape and Urban Planning*, Vol.91, No.3, 133–141, ISSN 0169-2046.

Harris, C. and Ullman, E. (1945). The Nature of Cities. *Annals of the American Academy of Political and Social Sciences*, Vol.242, No.1, 7-17, ISSN 0002-7162.

High Commission for Development of Riyadh (HCDR) (2004). *The Vision for the New. Suburban Cities, Executive Summary*. Arriyadh Development Authority, Riyadh, Saudi Arabia.

Hoyt, H. (1939). *The Structure and Growth of Residential Neighborhoods in American Cities. Federal Housing Administration*, Washington, D.C., US Government Printing Office.

Irwin, E.G., Jayaprakash, C. & Munroe, D.K. (2009). Towards a comprehensive framework for modeling urban spatial dynamics. *Landscape Ecology*, Vol.24, No.9, 1223–1236, ISSN 0921-2973.

Itami, R.M. (1988). Cellular worlds: models for dynamic conceptions of landscape. *Landscape Architecture*, Vol.78, No.5, 52-57, ISSN 0023-8031.

Jantz, C.A., Goetz, S.J., Donato, D. & Claggett, P. (2010). Designing and implementing a regional urban modeling system using the SLEUTH cellular urban model. *Computers, Environment and Urban Systems*, Vol.34, No.1, 1–16, ISSN 0198-9715.

Kim, B. (2005). *Modelling New City Growth Using Cellular Automata*. Unpublished PhD Thesis, School of Geography, University of Leeds, Leeds, UK.

Kirkby, M., Naden, P., Burt, T. & Butcher, D. (1992). *Computer Simulation in Physical Geography*. John Wiley and Sons, ISBN 0471 90604 2, Chichester.

Klosterman, R.E. (1999). The what if? Collaborative planning support systems. *Environment and Planning B*, Vol.26, No.3, 393-408, ISSN 0265-8135.

Knox, P.L. (1994). *Urbanization: An Introduction to Urban Geography*, Prentice-Hall, ISBN-10: 0131424505, NY.

Li, L., Sato, Y. & Zhu, H. (2003). Simulating spatial urban expansion based on a physical process. *Landscape Urban Planning*, Vol.64, No.1, 67-76, ISSN 0169-2046.

Li, X. & Yeh, A. (2001). Calibration of cellular automata by using neural networks for the simulation of complex urban system. *Environment and Planning A*, Vol.33, No.8, 1445–1462, ISSN 0308-518X.

Lavalle, C., Barredo, J.I., McCormick, N., Engelen, G., White, R. & Uljee, I. (2004). The MoLAND model for urban and regional growth forecast - A tool for the definition of sustainable development paths. European Commission, DG-Joint Research Centre, Ispra, Italy, 22 pp. EUR 21480 EN.

Ministry of Construction and Transportation (MOCT) (2005). Development Plan of Building New Administrative Capital. http://www.winwinkorea.go.kr.

Moreno, N., Wanga, F., Marceau & D.J. (2009). Implementation of a dynamic neighborhood in a land-use vector-based cellular automata model. *Computers, Environment and Urban Systems*, Vol.33, No.1, 44–54, ISSN 0198-9715.

Pacione, M. (2005). *Urban Geography: A Global Perspective, Second Edition*, Routledge, ISBN ISBN-10: 0415191963, London and New York.

Pan, Y., Roth, A.,Yu, Z. & Doluschitz, R. (2010). The impact of variation in scale on the behavior of a cellular automata used for land use change modelling. *Computers, Environment and Urban Systems*, Vol.34, No.5, 400–408, ISSN 0198-9715.

Petrova, L.O., Lavalle, C. & Kasanko, M. (2009). Urban land use scenarios for a tourist region in Europe: Applying the MOLAND model to Algarve, Portugal, *Landscape and Urban Planning*, Vol.92, No.1, 10–23, ISSN 0169-2046.

Poelmans, L. & Van Rompaey, A. (2009). Detecting and modelling spatial patterns of urban sprawl in highly fragmented areas: A case study in the Flanders–Brussels region. *Landscape and Urban Planning*, Vol.93, Nol.1, 10–19, ISSN 0169-2046.

Poelmans, L. & Van Rompaey, A. (2010). Complexity and performance of urban expansion models. *Computers, Environment and Urban Systems*, Vol.34, No.1, 17–27, ISSN 0198-9715.

Pontius, R.G., Cornell, J.D. & Hall, C.A.S. (2001). Modeling the spatial pattern of land-use change with GEOMOD2: application and validation for Costa Rica. *Agriculture, Ecosystems & Environment*, Vol.85, No.1, 191–203, ISSN 0167-8809.

Portugali, J. (2000). *Self-Organization and the City*. Springer-Verlag, ISBN 3-540-65483-6, Berlin.

Rafiee, R., Mahiny, A.S., Khorasani, N., Darvishsefat, A.A., Danekar, A. (2009). Simulating urban growth in Mashad City, Iran through the SLEUTH model (UGM). *Cities*, Vol.26, No.1, 19–26, ISSN 0264-2751.

Santé, I., Garcia, A.M., Miranda, D., Crecente, R. (2010).Cellular automata models for the simulation of real-world urban processes : A review and analysis. *Landscape and Urban Planning*, Vol.96, No.2, 108-122, ISSN 0169-2046.

Stevens, D. & Dragicevic, S. (2007). A GIS-based irregular cellular automata model of land-use change. *Environment and Planning B*, Vol.34, No.4, 708-724, ISSN 0265-8135.

Torrens, P.M. (2000a). *How Cellular Models of Urban Systems Work*. CASA Centre for Advanced Spatial Analysis, University College London, Working Paper 28, http://www.casa.ucl.ac.uk/publications/workingPaperDetail.asp?ID=28.

Torrens, P.M. (2000b). *How Land-Use Transport Models Work*. Centre for Advanced Spatial Analysis, University Collage London, Working Paper 20, http://www.casa.ucl.ac.uk/publications/workingPaperDetail.asp?ID=20.

van Vliet, J., White, R. & Dragicevic, S. (2009). Modeling urban growth using a variable grid cellular automaton. *Computers, Environment and Urban Systems*, Vol.33, No.1, 35-43, ISSN 0198-9715.

Veldkamp, A. & Fresco, L. O. (1996). CLUE: A conceptual model to study the conversion of land use and its effects. *Ecological Modelling*, Vol.85, No.2-3, 253–270, ISSN 0304-3800.

Weber, A. (1929) (translated by Carl J. Friedrich from Weber's 1909 book). *Theory of the Location of Industries*. The University of Chicago Press, ISBN-10: 0226264696, Chicago.

White, R., Engelen, G. & Uljee, I. (1997). Cellular automata as the basis of integrated dynamic regional modelling. *Environment and Planning B*, Vol.24, No.3, 323-343, ISSN 0265-8135.

United Nations (2010). *World Urbanization Prospects, The 2009 Revision: Press Release*. Department of Economic and Social Affairs, Population Division, New York.

Wolfram, S. (1994). *Cellular Automata and Complexity*. Addison-Wesley, ISBN 0-201-62716-7, Reading, MA.

Wu, X., Hu, Y., He, H.S., Rencang, B., Onsted, J. & Xi, F. (2009). Performance Evaluation of the SLEUTH Model in the Shenyang Metropolitan Area of Northeastern China. *Environmental Modelling and Assessment*, Vol.14, No.12, 21–230, ISSN 0167-6369.

Wu, D., Liu, J., Wang, S. & Wang, R. (2010). Simulating urban expansion by coupling a stochastic cellular automata model and socioeconomic indicators. *Stochastic Environmental Research and Risk Assessment*, Vol.24, No.2, 235–245, ISSN 1436-3240.

Wu, N. & Silva, E. (2009). Integration of genetic agents and cellular automata for dynamic urban growth modelling, *Proceedings of the 10th International Conference on GeoComputation*, University of New South Wales, Sydney, Australia, 30 Nov to 2 Dec 2009, Brisbane.

Yeh, A. & Li, X. (2001) A constrained CA model for the simulation and planning of the sustainable urban forms by using GIS. *Environment and Planning B,* Vol.28, No.5, 733-753, ISSN 0265-8135.

Zhang, J., Zhou, Y.K., Li, R.Q., Zhou, Z.J., Zhang, L.Q., Shi, Q.D. & Pan, X.L. (2010). Accuracy assessments and uncertainty analysis of spatially explicit modeling for land use/cover change and urbanization: A case in Beijing metropolitan area. *Science China Earth Sciences*, Vol.53, No.2, 173–180, ISSN 1674-7313.

6

Studies on Population Dynamics Using Cellular Automata

Rosana Motta Jafelice[1] and Patrícia Nunes da Silva[2]
[1]*Federal University of Uberlândia*
Faculty of Mathematics
[2]*State University of Rio de Janeiro*
Department of Mathematical Analysis - IME
Brazil

1. Introduction

In this chapter, we present three cellular automata that simulate the behavior of the population dynamics of three biological systems. They are shaped like a torus in which populations coexist artificially. The first one deals with artificially-living fish divided into two groups: sharks (predators) and fish that are part of their food chain (preys) (Edelstein-Keshet, 1988; Renning, 1999-2000). The second model introduces a simulation of the HIV evolution in the blood stream of positive individuals with no antiretroviral therapy (Jafelice et al., 2009). The last model extends the previous one and considers the HIV dynamics in individuals subject to medical treatment and the monitoring of the medication potency and treatment adhesion (Jafelice et al., 2009). For this purpose, a cellular automata approach coupled with fuzzy set theory is developed to study the HIV evolution. When modeling a physical or biological system there are many decisions to make. One of them is related to the kind of approach to take into account. In the "bottom-up" approach the complex behavior of a system emerges from the interaction of basic components. One might ask if it is possible to describe the behavior of complex systems in this way. As a matter of fact, Banks (1971) makes an interesting remark about how physicists were happy to believe the universe is composed of an enormous number of just three basic components: protons, electrons and neutrons. Modeling biological phenomena by realistic models generally leads to large system of non linear integro- and partial differential equations. One alternative approach is to consider cellular automata (CA) models. The usefulness of the CA models relays on simplicity and uniformity of their cells and also on their potential to model complex systems. The main idea behind cellular automata models is to consider each position (or region) of a spatial domain as a cell to which is attributed a certain state. The state of each cell is modified according to its own state and the states of its neighbor cells. These states are correlated through a number of simple rules that imitates the biological and physical laws that guide the system behavior (Ermentrout & Edelstein-Keshet, 1993). An important aspect in modelling population dynamics is to take into account the effect of the spatial distribution of population on their dynamics. The CA models can capture this effect (e.g. Czaran 1998; Lee et al. 1995).

2. Cellular automata of prey-predator model: Sharks and fish

In this section we discuss a cellular automaton for a predator-prey model proposed by David Wiseman at the University of Western Ontario (see Dewdney 1984). The planet Wa-Tor[1] is a grid shaped like a torus in which coexist artificially fish and sharks. In this planet most of the time fish and sharks move randomly. Fish and sharks can propagate when they have reached the appropriate age. Differently from sharks, fish have a plentiful supply of plankton and sharks have to eat fish in a specific maximal period, otherwise they would starve to death. Thus, the simulation is of a dynamic system of predator-prey type. In his two-dimensional CA, Dewdney (1984) considered a von Neumann neighborhood. That is, each cell is connected with itself and with its four orthogonal neighbors. He compared his results with the theoretical predator-prey relation given by the Lotka-Volterra equations and also with the sizes of the populations of the Canadian lynx and snowshoe hare recorded by the Hudson's Bay Company over almost fifty years. Our results were obtained using a two-dimensional CA with a Moore neighborhood. That is, each cell is connected with itself, with its four orthogonal neighbors and with its four diagonal neighbors. In both cases, just the earlier states of the cell and its neighbors determine the next time step. These kind of automata closely resembles an evolution equation such as partial differential or integral equations. Our results were compared to the Lotka-Volterra classic model (Edelstein-Keshet, 1988; Murray, 1990) and also with the empirical data from the Hudson's Bay Company.

2.1 Realistic example of predator-prey: Hare and lynx at Hudson's Bay

In 1850, the Hudson's Bay Company used to get from trappers pelts of hares and lynxes. The number of hares and lynxes that got into traps were recorded and used by researchers to study competitive interactions models (see Bulmer 1974; Stenseth et al. 1998). It is known that the number of the captured animals is proportional to their population, so researchers found populations statistics for those both species for over a large period. Data on the Canadian hare-lynx system based on the hare and lynx furs records of the Hudson's Bay Company may be found at Elton & Nicholson (1942), Gilpin (1973) and Hewitt (1921). Both populations do not exist independently from one another, because lynxes feed basically from hares. All the records of pelts for each year were analyzed by the ecologist Charles Elton (Elton, 1924) and are presented in the Fig. 1. The figure shows a regular oscillation over ten years in the numbers of both species. They change over 50 fold and up to 100 folds, over the cycle, what makes the amplitude of the oscillation huge. In the next subsections we will see that the classical model presents regular curves and a phase plane that is a perfect cycle while our cellular automaton will present a cycle which will be closer to the Fig. 2. The phase plane graph of a period of thirty years, initiated in 1875, was obtained from the populations data presented in the Fig. 1.

2.2 Classic predator-prey model

The population fluctuations of predators and prey challenged many researchers. What causes the changes in reproduction and survival? Besides the three main factors: food, predation, and social interactions, many other factors might affect these cycles. For instance, Zhang et al. (2007) used partial cross correlation and stepwise multiple regression methods to analyze the effect of climate on the Hudson's Bay Company's hare-lynx system (1847-1903). Their results showed that El Niño/Southern Oscillation has small effects on rates of increase in hare and lynx populations. Krebs et al. (2001) were interested in how changes on food can influence the

[1] The name Wa-Tor comes from Water-Toroidal.

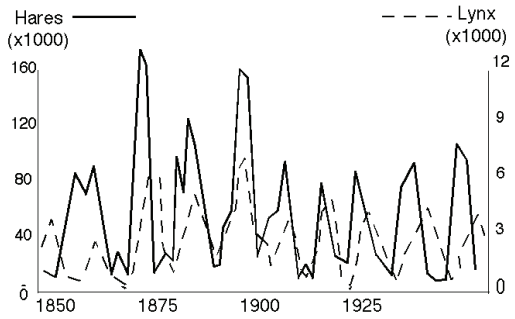


Fig. 1. Hares and lynxes population as a function of time (Renning, 1999-2000).

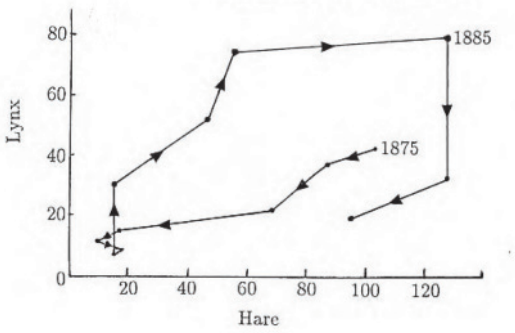


Fig. 2. Detail of the phase plane plot of the data presented in Fig.1 (Murray, 1990).

hare cycle while Stenseth et al. (1999) observed that lynx population dynamics are consistent with a regional structure caused by climatic features.
In order to formulate the interaction between preys and predators, a deterministic model (that became a classic model) was used[2]. It is given by the differential equations system:

$$\begin{aligned} \frac{dx}{dt} &= ax - \alpha xy \\ \frac{dy}{dt} &= -by + \beta xy \end{aligned} \tag{1}$$

In this model, the state variables x and y are, respectively, the number of preys and predators in each instant t. The parameters are:

- a: preys relative growth rate;
- α: predation rate (probability of a predator to kill the prey in each time they encounter);
- b predators mortality rate in the absence of preys;
- β preys conversion rate into predators.

Solving the differential equations with the parameters $a = 0.1$, $\alpha = 0.01$, $b = 0.05$ and $\beta = 0.001$, we obtain the graph of the Fig. 3 and the phase plane, Fig. 4.

[2] See Wangersky (1978) for a review on Lotka-Volterra population models.

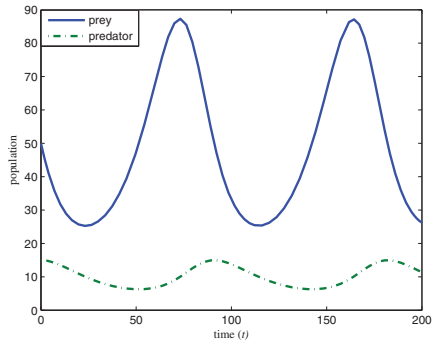


Fig. 3. Solution of the differential equation system.

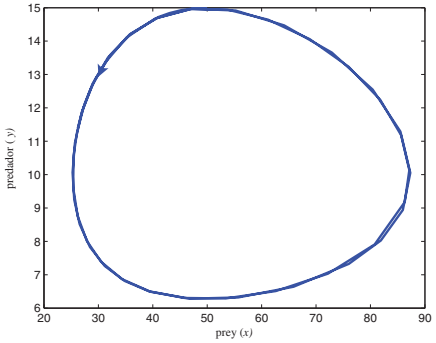


Fig. 4. Phase plane.

Since the classical model ignores spatial correlations it does not take into account important effects that spatial inhomogeneity may cause on the dynamics of the system. Moreover in such a model we do not have any information about the spatial distribution of the populations.

2.3 Description of the cellular automata simulation

The following five parameters need to be chosen to set up a simulation: 1. number of fish; 2. number of sharks; 3. fish reproductive age; 4. sharks reproductive age; 5. sharks starvation period. The initial number of sharks and fish as their respective ages are randomly distributed in a rectangular grid whose opposite sides are identified in pairs. The cells states in the grid are updated according to the local dynamics rules of each cell. For instance, in a 40x40 cell grid, 300 fish and 10 sharks are placed at random positions. All fish and sharks have a reproductive age, i.e., 3 and 10 iterations respectively and sharks starvation period is 3.

2.3.1 Behavior of fish in Wa-Tor

Each fish chooses a free place in its neighborhood, moves and ages there (if all places are occupied, then it remains where it is and ages). When it achieves the reproductive age, it leaves behind a single offspring. They move according to a randomly assigned integer that indicates a direction. More specifically, depending on whether the value of the integer is equal

to 1, 2, 3, 4, 5, 6, 7 or 8, they move north, east, south, west, northeast, northwest, southeast or southwest (Silva & Jafelice, 2010), in the grid, respectively.

2.3.2 Behavior of sharks in Wa-Tor

First, each shark searches for fish in its neighborhood. If there are fish, the shark randomly chooses one, catches it and the variable (starve variable) is set to zero. It goes to the cell of the eaten fish and might propagate if the occasion arises. If it does not find any fish in its neighborhood, it moves like a fish and the starve variable is increased by 1. When the starve variable reaches its maximum (sharks starvation period) the shark dies. Also, sharks reproduction is similar to the fish.

Simulation results of the Wa-Tor system for 200 iterations (or time steps) are depicted in Figs. 5-7. Notice that the behavior of the fish and sharks shown in Fig. 5 are close to the ones in similar phase shown in Fig. 3.

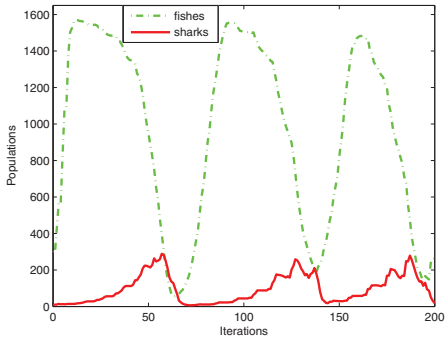


Fig. 5. Wa-Tor simulation results.

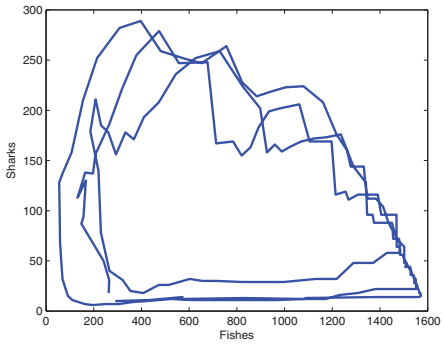


Fig. 6. Phase plan for a simulation in Wa-Tor.

2.4 Computational graphical interface

We have used *Matlab 7.0* to build our computational graphical interface for the Wa-Tor system (see its initial interface in Fig. 8 – (Jafelice & Silva, 2001)). To set up your own simulation,

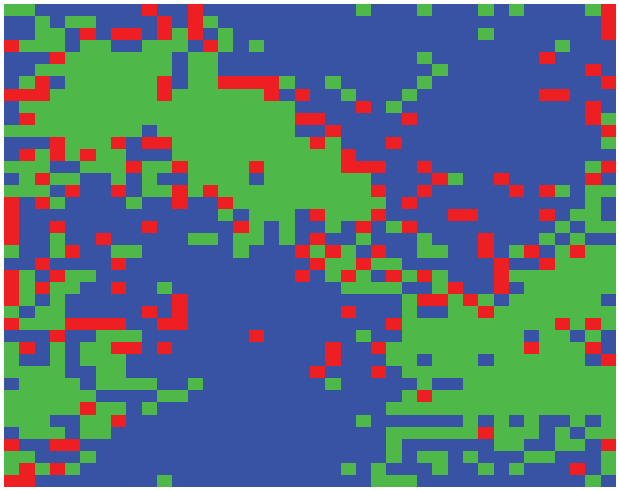


Fig. 7. Snapshot of the cellular automata model output: the blue background is the sea, the fish are in green and the sharks in red.

the following six parameters need to be chosen: 1. number of iterations 2. initial number of fish; 3. initial number of sharks; 4. fish reproductive age; 5. sharks reproductive age; 6. sharks starvation period. It is also possible to run the four simulations listed below with previously assigned parameters:

1. stable ecological cycle
2. fish extinction
3. shark extinction
4. fish and shark extinction.

Furthermore, at the end of each simulation, the graphs of fish and sharks population as a function of time and of the phase plane are plotted.

2.5 Conclusion

This section has introduced a cellular automata approach to a prey-predator dynamics. The obtained results (as well as the Dewdney (1984) ones) resemble the Lotka-Volterra ones but go further. The population fluctuations of fish and shark resemble better the hare and lynx charts than the Lotka-Volterra solutions do. Another interesting feature of the CA model is that it is a spatially distributed prey-predator model. Nowadays, the crucial role of spatial inhomogeneity into the dynamics of biological species has been recognized (see Durrett & Levin (2000), Ermentrout & Edelstein-Keshet (1993) and the references therein for further discussion. See also, Pekalski (2004) for an overview on predator-prey systems approaches and remarks on some open problems). Saila (2009) presents Wa-Tor as a complex adaptive system of interacting autonomous agents and points out that the utility of conventional mathematics in understanding the dynamics of such complex ecosystems is limited. The evolution of the

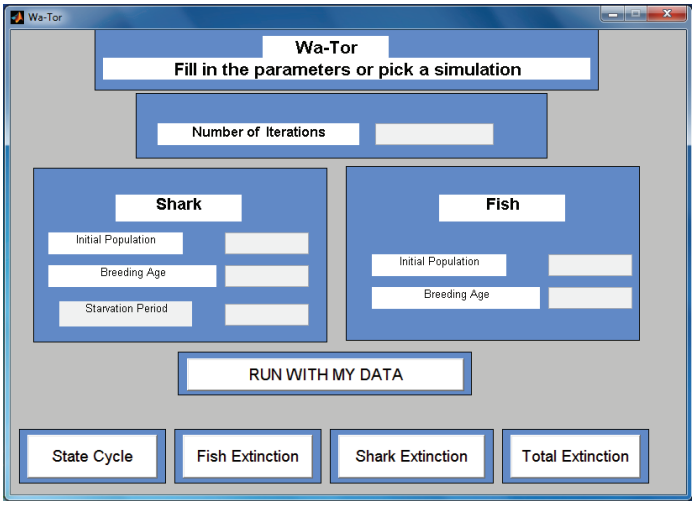


Fig. 8. Computational Graphical Interface of Wa-Tor.

system does not seem to depend on the initial random distribution but the choice of the five initial parameters is a critical point for the future behavior of the system. The model presents a complex behavior and the simulations give a qualitative image of the reality.

Cellular automata models for the Human Immunodeficiency Virus (HIV) infection dynamics are the subject of the next sections. Deciding which are the dynamics rules of each cell demands a deep knowledge on the HIV behavior. HIV is a spherical retrovirus composed of RNA or ribonucleic acid. Its replication occurs within host cells. Three virus proteins are of paramount importance for the replication process: Reverse Transcriptase, Integrase and Protease. After HIV gains entry to its human host, it is disseminated throughout the lymphatic tissues. When HIV reaches the blood stream, it attacks mainly the lymphocyte T of the $CD4+$ type. The quantity of cells $CD4+$ in periphery blood has prognostic implications in HIV infection evolution. The gradual loss of CD4+ T cells to the AIDS-defining level of 200 cells/mm^3 and progressive immune deficiency lead to opportunistic infections that characterizes the HIV infection (Haase, 1999; Hazenberg et al., 2000). Nowadays, the amount of immunocompetent cells is the most clinically used and acceptable measurement during treatment of infected individuals. The antiretroviral treatment works inhibiting both reverse transcriptase and protease. The inhibitions of reverse transcriptase prevents free virus particles to infect $CD4+$ cells. Protease inhibition delays the viral replication, allowing the organism to react naturally. Combination of reverse transcriptase and protease inhibition has led to a substantial improvement in HIV therapy.

Microscopic models for HIV infection dynamics in human individuals provide helpful information to construct cellular automata models, especially when the growth rate of T lymphocyte of $CD4+$, the death rate of infected and non infected cells, free virus load, specific antibodies CTL, interaction rate between non infected cells of the T $CD4+$ and the virus, and the interaction rate between the infected cells of the lymphocyte T of the $CD4+$ and the antibody are constant.

In the next section, we introduce the cellular automaton model for the HIV infection dynamics with no antiretroviral therapy (Jafelice et al., 2009).

3. Cellular automata of the HIV evolution in the blood stream of positive individuals with no antiretroviral therapy

AIDS (Acquired Immunodeficiency Syndrome) has become a worldwide health problem. In countries where AIDS control is poor or even nonexistent, as in some African nations, the HIV-positive population shows high mortality rates. Zorzenon dos Santos & Coutinho (2001) reported a cellular automaton approach to simulate the three-phases patterns of HIV infection consisting of primary response, clinical latency and onset of acquired immunodeficiency syndrome (AIDS). The robustness of the results obtained from their cellular automata model were analyzed in Figueiredo et al. (2008). The CA model from Ueda et al. (2006) considers the diversity exhibited by both HIV and T cells. Their results indicate the diversity of the virus is the major factor affecting the success rate of the escape of HIV from the immune response and they were also able to resemble the incubation time variability observed in vivo. Mielke & Pandey (1998) developed a fuzzy interaction model for mutating HIV with a fuzzy set of 10 interactions for macrophages, helper cells, cytotoxic cells and virion. These models are cellular automata models with no antiretroviral treatment. We consider a cellular automaton to model the behavior of the three-phases pattern of HIV infection which consists of: primary infection, asymptomatic and symptomatic phases for the cells of T lymphocyte of $CD4+$ of the HIV and specific antibodies called CTL. We compare our CA model results with the natural history of HIV infection and also with the HIV dynamics model proposed by Nowak & Bangham (1996).

3.1 Microscopic models of HIV dynamics

Nowak & Bangham (1996) developed three microscopic models for HIV infection dynamics within the organism of human individuals, considering no antiretroviral therapy.
The first model captures the interaction between replicating virus and host cells. In this case, three variables are considered: uninfected cells n, infected cells i and free virus particles v. This model assumes that infected cells are produced from uninfected cells and free virus at rate βnv and die at rate bi. Free virus is produced from infected cells at rate ki and declines at rate sv. Uninfected cells are produced at a constant rate, r, from a pool of precursor cells, and die at rate an. Fig. 9 illustrates the HIV dynamics developed by this model.

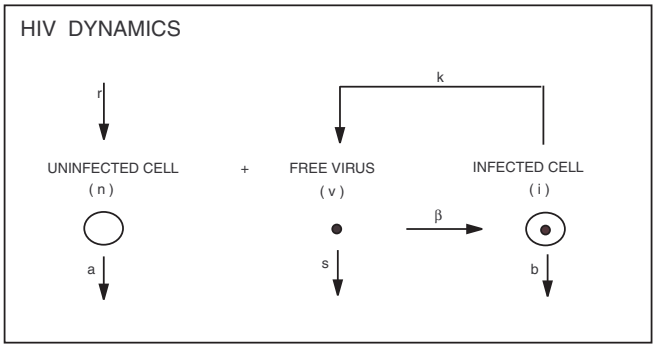


Fig. 9. Microscopic models HIV virus dynamics (Nowak, 1999).

Modeling assumptions lead to the following system of differential equations:

$$\begin{aligned} \frac{dn}{dt} &= r - an - \beta nv \\ \frac{di}{dt} &= \beta nv - bi \\ \frac{dv}{dt} &= ki - sv. \end{aligned} \tag{2}$$

The second model includes immune responses against infected cells, and extends the system of equations (2) adding an equation to describe the immune responses against infected cells:

$$\begin{aligned} \frac{dn}{dt} &= r - an - \beta nv \\ \frac{di}{dt} &= \beta nv - bi - piz \\ \frac{dv}{dt} &= ki - sv \\ \frac{dz}{dt} &= ciz - dz. \end{aligned} \tag{3}$$

The variable z denotes the magnitude of the antibodies CTL (cytotoxic T lymphocyte) – that is, the abundance of virus-specific CTLs. The rate of CTL proliferation in response to antigen is ciz. In the absence of stimulation, CTLs decay at rate dz. Infected cells are killed by CTLs at rate piz. Fig. 10 shows the solution of (3) using the parameters of Table 1 and initial conditions of Table 2, obtained from Caetano & Yoneyama (1999).

$r = 0.3$	$a = 0.1$	$\beta = 1$
$b = 0.01$	$p = 0.03$	$k = 0.5$
$s = 0.01$	$c = 0.01$	$d = 0.01$

Table 1. Parameters of the microscopic HIV model.

$n(0)$	0.99
$i(0)$	0.01
$v(0)$	0.1
$z(0)$	0.01
t initial	0
t final	500 time units

Table 2. Initial conditions of the microscopic HIV model.

From Fig. 10 one can see that, in logarithmic scale, the uninfected cells of $CD4+$ show a rapid decline in the first weeks and a slow recovery when the number of lymphocytes is close to the maximum. The increase in the number of lymphocytes is related to virus replication in the infected cells.

Comparing the solution of system (3), shown in Fig. 10, with the plots of Fig. 11, which gives the dynamics of HIV infection history currently accepted (Coutinho et al., 2001; Perelson & Nelson, 1999; Saag, 1995), we notice that the uninfected cells of $CD4+$ identify with the $CD4+$ level, the free virus with the HIV virus, and the virus-specific CTLs with the HIV

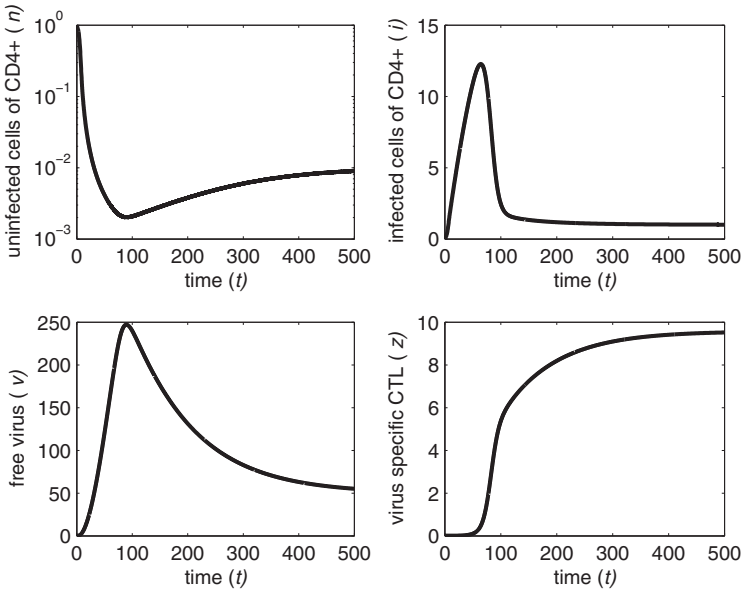


Fig. 10. Solution of the microscopic HIV model (3).

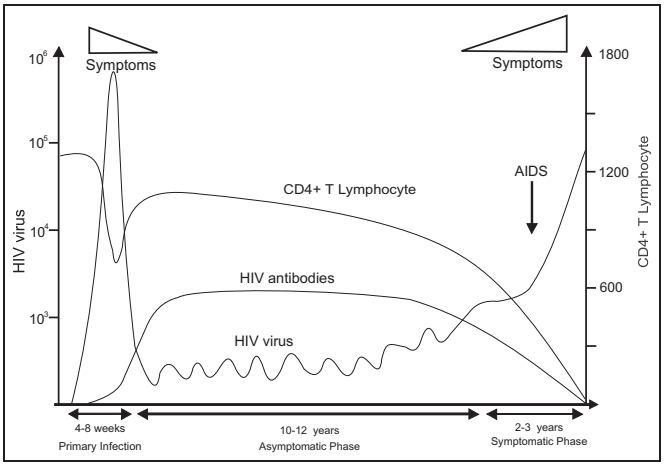


Fig. 11. The natural history of HIV infection dynamics as currently accepted (Coutinho et al., 2001; Perelson & Nelson, 1999; Saag, 1995).

antibodies. This result will help to validate the cellular automata model for individuals under no antiretroviral therapy.

The next section details the Blood-Tor system, the cellular automaton model for the HIV infection dynamics.

4. Blood-Tor system

The name Blood-Tor comes from Bloodstream-Toroidal which is similar to the name of the cellular automaton Wa-Tor, that means Water-Toroidal (Dewdney, 1984; Renning, 1999-2000). The Blood-Tor System (BTS) is shaped like a torus in which coexist artificially uninfected cells, infected cells of lymphocytes T of $CD4+$, free virus particles, and specific antibodies CTL (cytotoxic T lymphocyte) that attack infected cells in an individual blood stream with no antiretroviral therapy. These elements are the same as those associated with the state variables of the differential equation system (3).

4.1 Description of the simulation process

Eleven parameters need to be chosen to set up a simulation run. The parameters are the following:

- number of uninfected cells
- number of infected cells of the lymphocytes T of $CD4+$
- number of free virus particles
- number of specific antibodies CTL (cytotoxic T lymphocyte)
- life span limit of uninfected cells
- life span limit of infected cells
- life span limit of free virus particles
- life span limit of specific antibodies CTL (cytotoxic T lymphocyte)
- infected cells reproductive age
- specific antibodies CTL reproductive age
- uninfected cells production rate

The cells states in the grid are updated according to the local dynamics rules of each cell. For instance, in a 31x31 cell grid 200 uninfected cells, 16 infected cells of the lymphocytes T of $CD4+$, 120 free virus particles and 25 specific antibodies CTL (cytotoxic T lymphocyte) are placed at random positions. All uninfected cells, infected cells of the lymphocytes T of $CD4+$, free virus particles and specific antibodies CTL have a life span set according to a specific time limit. Table 3 gives the values of the life spans. An initially random assortment of ages are distributed to the elements (uninfected cells, infected cells of lymphocytes T of $CD4+$, free virus particles and specific antibodies CTL that attack infected cells) according to their respective life span limits.

cell	uninfected	infected	HIV	CTL
life span limit (iterations)	4	5	3	15

Table 3. Life span limits.

4.1.1 Behavior of uninfected cell of the lymphocytes T of $CD4+$ in BTS

Each uninfected cell of the lymphocytes T of $CD4+$ chooses a free place in its neighborhood, moves and ages there (if all places are occupied, then it remains where it is and ages). They move according to a randomly assigned integer that indicates a direction. More specifically, depending on whether the value of the integer is equal to 0, 1, 2 or 3, they move north, east,

south or west in the grid, respectively. Lymphocytes T of $CD4+$ are produced with a constant rate. During simulation the rate is 18 cells for each iteration. When they reach their life span limit they die.

4.1.2 Behavior of HIV in BTS

First, each HIV searches for uninfected cells of the lymphocytes T of $CD4+$ in its neighborhood. If there are uninfected cells, the HIV randomly chooses one and the cell chosen becomes an infected cell. If there are no uninfected cells, then the HIV chooses a place in its neighborhood and moves and ages there (if all places are occupied, it remains in its place and ages). When HIVs reach their life span limit they die.

4.1.3 Behavior of infected cell of the lymphocytes T of $CD4+$ in BTS

When free HIVs encounter uninfected cells of $CD4+$, the uninfected cells become infected. Those cells begin to replicate HIV when they reach the age of 5 iterations. The simulation program puts a HIV in the position of the infected cell and assigns zero age to the new HIV. They move and age similarly as the uninfected cells lymphocytes T of $CD4+$. After their life span limit, they die.

4.1.4 Behavior of specific antibodies CTL in BTS

Each specific antibody CTL looks for infected cells of the lymphocytes T of $CD4+$ in its neighborhood. When specific antibodies CTL encounter infected cells, the infected cells are destroyed. The specific antibodies CTL move to the cells infected in previous position. The specific antibodies reach the reproduction period after 14 iterations. They move and age similarly as the uninfected cells of lymphocytes T of $CD4+$. After their life span limit, they die.

The Blood-Tor system simulates the dynamics of the evolution of HIV within the blood stream human individual with no treatment.

Fig. 12 shows a snapshot of the Blood-Tor system cellular automaton model. The uninfected cells are shown in blue, the HIVs in black, the infected cells in green, and the antibodies in white. The Blood-Tor simulation system was developed using Matlab 7.0.

Simulation results of the Blood-Tor system (BTS), after 50 iterations (or time steps) are depicted in Fig. 13. Notice that the behavior of the uninfected cells of $CD4+$, infected cells of $CD4+$, free virus, and virus specific antibodies shown in Fig. 13 are close to the ones in similar (asymptomatic) phase shown in Fig. 11. Clearly, BTS does give a good description of the evolution of HIV in the blood stream of human individuals with no treatment.

Next section extends the BTS to encompass the natural phases of HIV dynamics of the Fig. 11.

4.2 Extended Blood-Tor system

To expand the ability of cellular automaton to model the natural history of HIV infection, the Blood-Tor System was extended to include the symptomatic phase behavior, as Fig. 11 suggests.

The cellular automaton that produces the outputs shown in Fig. 13, was modified such that, after a certain number of iterations, antibodies production decrease and, consequently, the number of free virus particles increases. In a grid of 31x31 cells, 120 uninfected cells, 18 infected cells of the lymphocytes T of $CD4+$, 180 free virus particles and 18 specific antibodies CTL (cytotoxic T lymphocyte) were introduced at random positions. All of these cells move

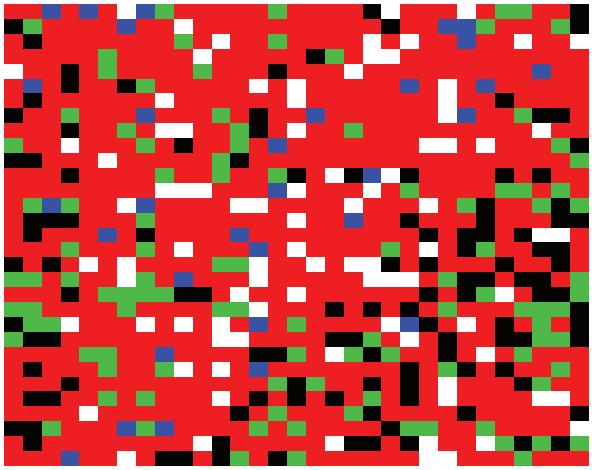


Fig. 12. Snapshot of the cellular automata model output: the red background is the blood stream, the uninfected cells are in blue, the HIV in black, the infected cells in green, and the antibodies in white.

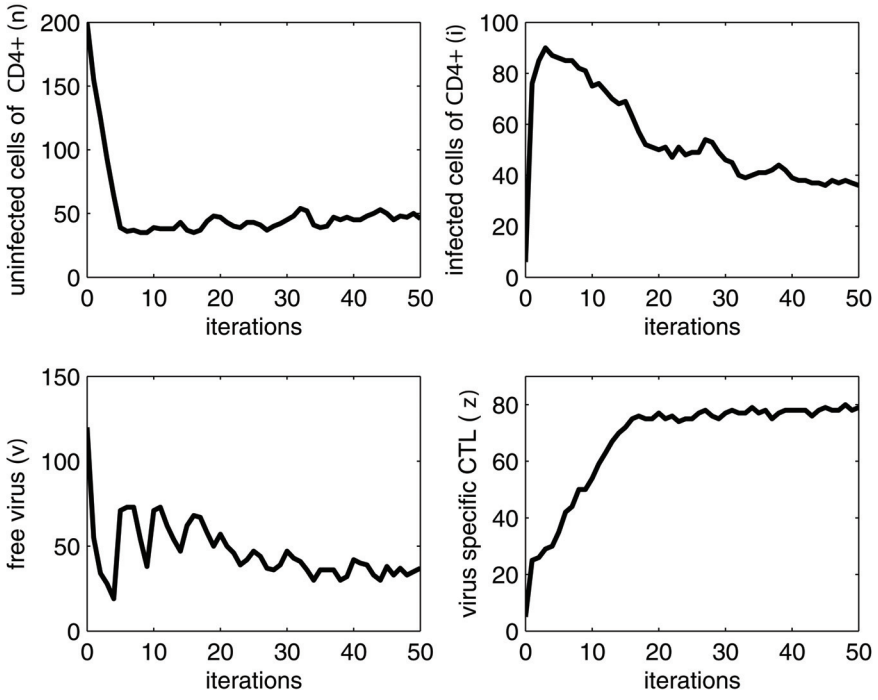


Fig. 13. Blood-Tor system simulation results.

randomly. The time limit of uninfected cells was set to seven (in the previous case it was set at four iterations). If the number of iterations is smaller than 70, then the reproduction time can be chosen as 14 iterations. If the number of iterations is greater than or equal 70, then the reproduction time decreases during the next iterations. If the number of iterations is greater than 90, then the number of uninfected cells placed at each iteration decreases. The result of this choice reflects the failure of the immunological system. That is, the immune system of the human individual looses the capacity to fight the viruses.
The BTS simulation results become, in this case, very close to actual HIV biological dynamics. They show strong qualitative similarities during all the natural history of HIV infection dynamics, as Figs. 11 and 14 clearly show.

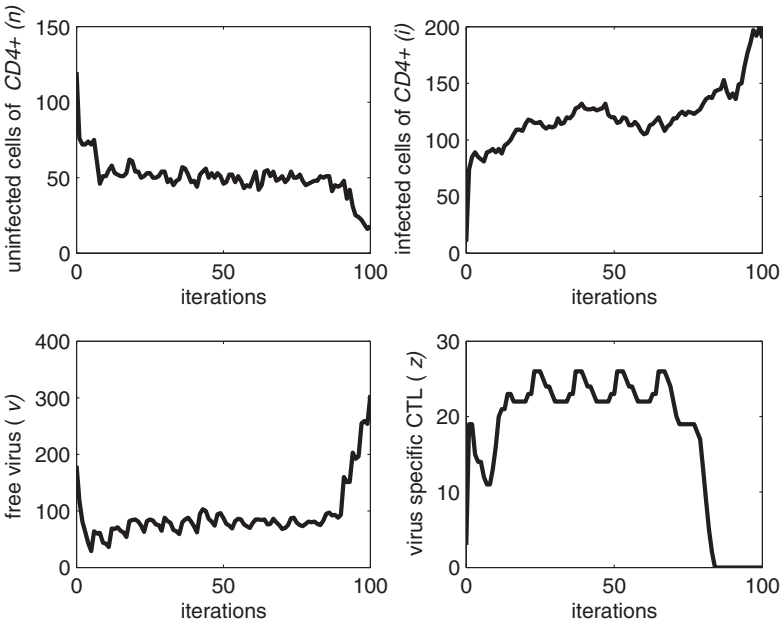


Fig. 14. Extended Blood-Tor system simulation results.

It is well known that AIDS is a disease that can be treated using appropriate drugs, but no absolute cure mechanism has been found yet. Some antiretroviral therapy uses reverse transcriptase inhibitors, others fight against an enzyme that is essential for the formation of infectious virus particles from infected cells called viral protease. All anti-HIV drugs aim at preventing the virus from reproducing, but they do not kill virus particles or infected cells (Nowak, 1999). Inspired in Zorzenon dos Santos & Coutinho (2001) CA model, Sloot et al. (2002) proposed a CA model incorporating drug therapy. Its main ingredients are destruction of previously emerged spatial patterns (wave-like and solid-like structures) and reconstruction of new spatial patterns (wave-like structures) due to incorporation of the drug therapy concept. The CA model integrates three different therapy procedures into one model and the simulations show a qualitative correspondence to clinical data. Shi et al. (2008)

presented a CA model for HIV dynamics and drug treatment. It includes the virus replication cycle and mechanisms of drug therapy. Viral load, its effect on infection rate, and the role of latently infected cells in sustaining HIV infection are among the aspects that are explored and incorporated in the model. The dynamics from the model qualitatively match clinical data. In the next section, we present the cellular automaton model for the HIV infection dynamics with antiretroviral therapy (Jafelice et al., 2009).

5. Cellular automata of the HIV evolution in the blood stream of positive individuals with antiretroviral therapy

The Blood-Tor System, detailed in section 4, simulates the behavior of HIV infection dynamics in the blood stream of HIV positive human individuals who have not received any antiretroviral therapy. This section addresses the Bloodstream-Toroidal system when treatment is taken into account. Its purpose is to model and simulate the HIV dynamics in the blood stream of individuals subject to antiretroviral therapy.

To simulate the antiretroviral therapy the BTS system adopts fuzzy parameters due the imprecise nature of how the individuals respond to the antiretroviral therapy. When accounting for antiretroviral therapy, the cellular automaton model assumes that the viruses do not infect all $CD4+$ cells because only a portion of $CD4+$ cells are usually infected. The period of virus replication is delayed, similarly as it happens in positive HIV individuals blood stream. The fuzzy parameters depend on the medication potency and on the adhesion of the individuals to the treatment. Adhesion to treatment means how individuals follow the correct medication prescription of the therapy. Adhesion is a very complex issue because it involves many factors that affect the ability of the individuals to comply with the antiretroviral therapy. Many factors can interfere in the regime prescribed, including the number of hours that individuals sleep, how strict they are with meals, medication schedules and how healthy their social life is. Information about medication potency can be obtained from medical doctors using their knowledge from clinical trials, clinical experience and knowledge published in the relevant literature. Along with clinical experience, the model increases the $CD4+$ level and decreases the viral load to simulate the antiretroviral therapy. The next subsection briefly review the concept of fuzzy set and fuzzy rule-based systems.

5.1 Basic concepts of fuzzy set theory

The literature on uncertainty has grown considerably during these last years, especially in the areas of system modeling, optimization, control, and pattern recognition. Recently, several authors have advocated the use of fuzzy set theory to address epidemiology problems (Barros et al., 2003; Jafelice et al., 2004; 2005; Ortega et al., 2003) and population dynamics (Krivan & Colombo, 1998). Since the advent of the HIV infection, several mathematical models have been developed to describe the HIV dynamics (Murray, 1990; Nowak & Bangham, 1996; Nowak, 1999). Here, we suggest the use of fuzzy set theory (Zadeh, 1965) to deal with the uncertain, imprecise nature of the virus dynamics.

First, we recall that a fuzzy set A on a universal set X is a membership function A that assigns to each element x of X a number $A(x)$ between zero and one to indicate the degree of membership of x in A. Therefore, the membership function of the fuzzy set A is a function $A : X \to [0,1]$. It is interesting to note that a conventional set A on X is a particular instance of a fuzzy set for which the membership function is the characteristic function of A, that is, $\mathcal{X}_A : X \to \{0,1\}$.

Second, we remind the reader that a concept that plays a key role in fuzzy set theory is fuzzy

rule-based systems (FRBS) (Pedrycz & Gomide, 1988). The structure of FRBS is shown in Fig. 15.

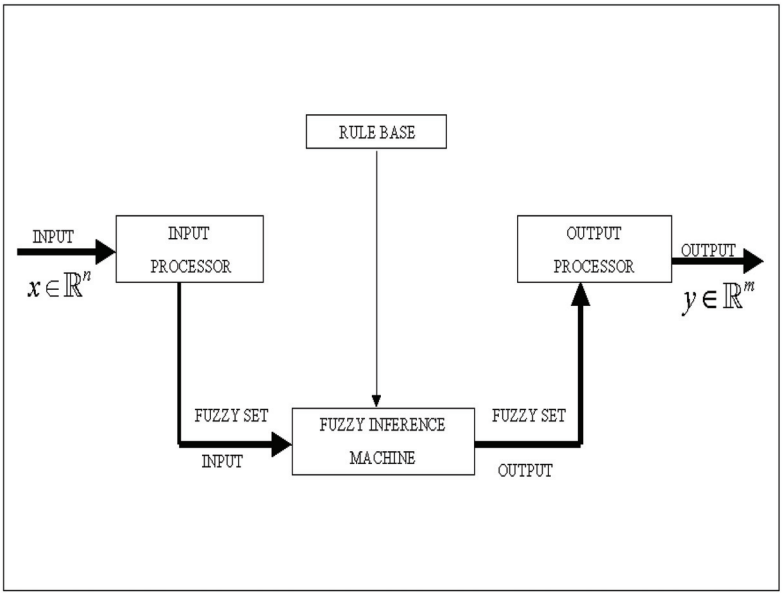


Fig. 15. Structure of fuzzy rule-based systems.

A FRBS has four components: an input processor, a collection of fuzzy rules called fuzzy rule base (or rule base for short), a fuzzy inference machine, and an output processor. These components process real-valued inputs to provide real-valued outputs as follows.

- **Input Processor (Fuzzification).** Here, inputs are encoded into fuzzy sets on the respective universes of the input variables. For numerical inputs, the approach commonly used is to transform a real-valued input into a fuzzy singleton. Expert knowledge plays an important role to build the membership functions for each fuzzy set associated with the inputs.
- **Rule Base.** This a knowledge-encoding component of fuzzy rule-based systems, a collection of fuzzy conditional propositions in the form of If-then rules. Fuzzy rules are an effective mean to encode expert knowledge expressed through linguistic statements. In general, If-then rules describe relationships between linguistic variables such as If *adhesion to treatment* is *low* and *medication potency* is *high* then *period of virus reproduction* is *fast* and *percentage of infected CD4+ cells* is *high*. In fuzzy set theory, a variable (e.g. *adhesion to treatment*) whose value is a linguistic term (e.g. *low*) is called a linguistic variable (Pedrycz & Gomide, 1988).
- **Fuzzy Inference.** The fuzzy inference machine performs approximate reasoning using the compositional rule of inference. A particular form of fuzzy inference that is of interest in this paper is the Mamdani method (Mamdani, 1976; Mamdani & Assilian, 1999), derived from the max-min composition (Pedrycz & Gomide, 1988).

- **Output Processor (Defuzzification).** In fuzzy rule-based systems, the inferred output usually is a fuzzy set. Often, especially in biological systems modeling, we require a real-valued output. The output processor task is to provide real-valued outputs using defuzzification, a process that chooses a real number that is representative of the fuzzy set inferred. A typical defuzzification scheme, the one adopted in this paper, is the centroid or center of mass method (Jafelice et al., 2004).

5.2 Linguistic variables and rule base

Fuzzy set theory is a mathematical tool to model imprecise information and knowledge. In practice, precise values of the number of infected $CD4+$ cells and the period of virus replication with the antiretroviral therapy is uncertain. These values depend on the medication potency and on the individuals adhesion to treatment. Fuzzy rule-based systems (FRBS) is an appropriate approach to address the effect of the treatment in HIV dynamics. The input variables of the FRBS are the adhesion to treatment and the medication potency (Ying et al., 2007). The output variables are the percentage of HIV infected $CD4+$ cells and the period of virus replication. The input and output variables are linguistic variables, denoted as A, M, P and V. Adhesion to treatment (A), medication potency (M) and percentage of infected $CD4+$ cells (P) assume the following linguistic values {*very low*, *low*, *medium*, *high*, *very high*} and the period of virus replication (V) adopts the linguistic values {*very rapid*, *rapid*, *medium*, *slow*, *very slow*}. The membership functions specify the meaning of the linguistic variables, as depicted in Figs. 16, 17, 18 and 19 for *adhesion to treatment*, *medication potency*, *period of virus replication*, and *percentage of CD4+ cells that will be infected*, respectively. The rule base that encodes the relationships between A, M, P and V was constructed using expert medical knowledge. The fuzzy rules are summarized in Tables 4 and 5. The rule base was processed using the Mamdani inference method with centroid defuzzification.

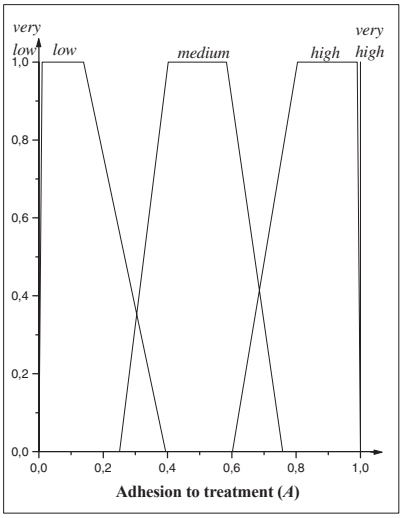


Fig. 16. Membership functions for adhesion to treatment (A).

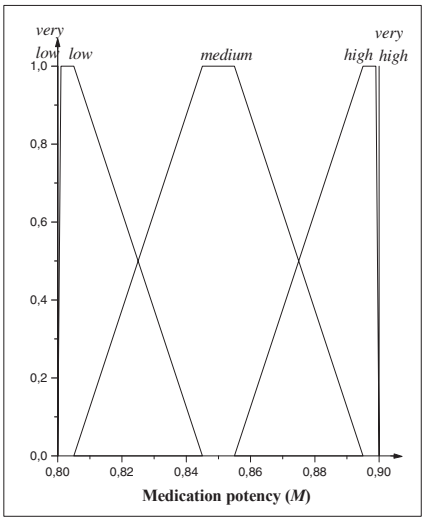


Fig. 17. Membership functions for medication potency (M).

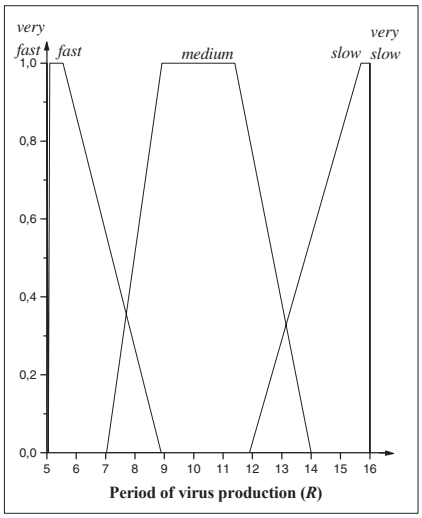


Fig. 18. Membership functions for period of virus replication (R).

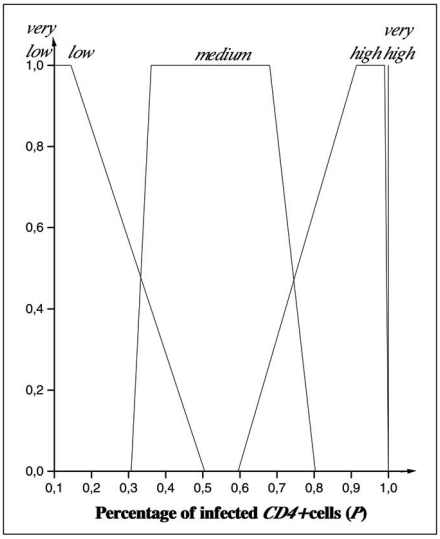


Fig. 19. Membership functions for percentage of $CD4+$ cells that infected (P).

Medication Potency (M) / Adhesion(A)	*very low*	*low*	*medium*	*high*	*very high*
very low	*very high*	*very high*	*very high*	*very high*	*very high*
low	*high*	*high*	*high*	*high*	*high*
medium	*medium*	*high*	*medium*	*high*	*medium*
high	*medium*	*high*	*low*	*low*	*low*
very high	*low*	*low*	*low*	*very low*	*very low*

Table 4. Fuzzy rules for the percentage of $CD4+$ cells that will be infected.

Medicat. Potency (M) / Adhesion(A)	*very low*	*low*	*medium*	*high*	*very high*
very low	*very rapid*	*very rapid*	*very rapid*	*very rapid*	*very rapid*
low	*rapid*	*rapid*	*rapid*	*rapid*	*rapid*
medium	*medium*	*rapid*	*medium*	*rapid*	*medium*
high	*medium*	*rapid*	*medium*	*slow*	*slow*
very high	*rapid*	*medium*	*medium*	*very slow*	*very slow*

Table 5. Fuzzy rules for the period of virus replication.

The cellular automaton model uses the output of the FRBS as follows. The number of HIV in the neighborhood of uninfected $CD4+$ cells is counted at each iteration. The product of the number counted multiplied by the output variable (percentage of $CD4+$ cells) is the number of HIV infected cells. This operation models the action of reverse transcriptase inhibitors. The percentage of $CD4+$ cells depends on the adhesion to treatment and on the potency of the medication. All those processes occur at each iteration. In the cellular automaton representing HIV infection dynamics and untreated HIV positive individuals, an infected cell $CD4+$ releases one virus at an available free place of its neighborhood after 5 iterations. In the

cellular automaton with treatment, the period of virus replication varies from 5 to 16 iterations, which models inhibitors action in delaying viral replication.

6. Simulation of the blood-tor system with treatment

6.1 Analysis of the Solutions

The quantity and specific time limit of uninfected cells, infected cells of the lymphocytes T of $CD4+$, free virus particles and specific antibodies CTL (cytotoxic T lymphocyte) were adjusted for different patients, considering their adhesion to the treatment. Simulation was carried out using the data (treatment adhesion and medication potency) of three HIV positive individuals shown in Table 6. In the table, the parameters of the first, second and third columns correspond to HIV positive individuals whose treatment receives low, medium, and high potency medication and treatment, respectively. The output variable values of the fuzzy rule-based system are shown in Table 7. The first line of the table shows the percentage of the $CD4+$ cells infected, and the second shows the period of virus replication, for the input values of Table 6. The cellular automaton model was ran five times for each patient. Averages are computed at each time instant t. Fig. 20 shows the results. The average of the individuals with the best response to the treatment is depicted in solid line. The dotted line is the average of the individuals with the worst response to the treatment. The behavior of the HIV as well as the behavior of the uninfected cells of type T lymphocyte of $CD4+$ fully agree with the corresponding behaviors reported in Guedj et al. (2007), Filter et al. (2005) and Ouattara et al. (2008). The HIV curve exhibits an asymptotic decay with a positive upper bound. In practice, laboratory exams may not detect the viral load, but indicate that the number of RNA copies of the virus in blood circulation is below the precision of the method used. The precision values are variable. For instance, in the case of the Brazilian public health network, the method currently adopted has the precision of 50 copies$/ml$ (Brazil, 2008). If a patient does not adhere to the treatment, then simulation proceeds as in the no treatment case discussed in section 4.

	First parameter	Second parameter	Third parameter
Medication potency	0.8	0.85	0.9
Adhesion to treatment	0.1	0.6	1

Table 6. Inputs for the FRBS used in simulation.

	First parameter	Second parameter	Third parameter
Percentage of $CD4+$ cells infected	0.85	0.55	0.1
Period of virus replication	6.35	10.37	16

Table 7. Outputs of the FRBS used in simulation.

6.2 Treatment response estimation

Fig. 20 suggests that the treatment response of patient p_s is better than of patient p_f, where p_f and p_s are the data of patients corresponding to first and second parameters of Table 6, respectively. To quantify the treatment response we must define a performance measure. Any of the four (or all, if an appropriate temporal average is chosen) variables, namely, uninfected and infected cells of lymphocyte T $CD4+$, free virus, and specific antibodies, can be considered. For instance, an estimation can be obtained using the a ratio of two variables values. Let us assume that the viral load is the variable revealing the treatment efficiency. Let $\overline{v_{p_1}}$ and $\overline{v_{p_2}}$ the averages of the viral loads of the patient 1 and 2 over the same time interval

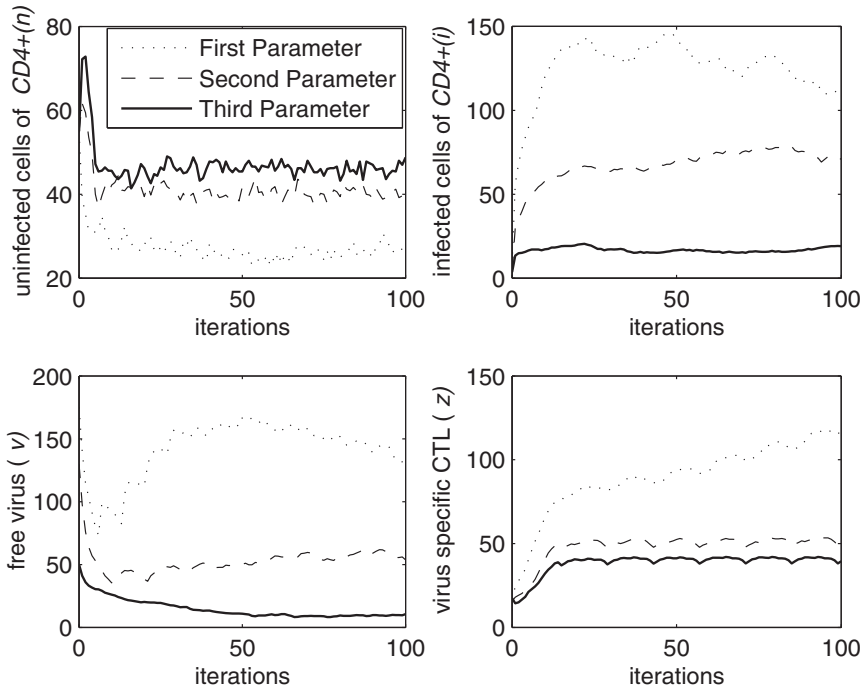


Fig. 20. Averages of the Blood-Tor System with Treatment beginning at $t = 0$.

$\triangle t$. The treatment response (C_r) in terms of the ratio of the averages $\overline{v_{p_1}}$ and $\overline{v_{p_2}}$ signalizes the treatment efficiency. Therefore, we have

$$C_r = \frac{\overline{v_{p_1}}}{\overline{v_{p_2}}} = \frac{\frac{\sum v_{p_1}}{\triangle t}}{\frac{\sum v_{p_2}}{\triangle t}} = \frac{\sum v_{p_1}}{\sum v_{p_2}}. \tag{4}$$

To illustrate the use of (4) with $\triangle t = 100$ we obtain $\overline{v_{p_f}} = 139.24$ and $\overline{v_{p_s}} = 50.71$. The averages $\overline{v_{p_f}}$ and $\overline{v_{p_s}}$ were computed using the outputs of the cellular automaton. Thus, the response ratio between the first and second patient parameters is

$$C_{r_{fs}} = \frac{\overline{v_{p_f}}}{\overline{v_{p_s}}} = \frac{139.24}{50.71} = 2.74.$$

The remaining cases are similar. Notice that, for the example just discussed, patient p_s response is twice as better than patient p_f. The values of $\overline{v_{p_f}}$ and $\overline{v_{p_s}}$ are averages of 30 runs of the model.

6.3 Conclusion

The sections 4 and 5 introduced a cellular automata approach to model HIV positive behavior in two cases: without and with antiretroviral treatment. An interesting characteristic of the

model is its ability to approximate the trajectories of all phases of the HIV history. Most models suggested so far emphasize the asymptomatic phase only. Moreover, the similarity of the solutions of the cellular automata models with the natural history (Fig. 11) gives enough evidence that they reproduce the actual HIV dynamics appropriately. The Blood-Tor System with treatment taken into account approximates the dynamics of HIV infection in the blood stream of HIV positive individuals under antiretroviral therapy. The outputs of the fuzzy rule-based system provide the percentage of infected $CD4+$ cells and the period of virus replication, given information about medication potency and individual adhesion to treatment. Using the outputs of the fuzzy rule-based system, the cellular automaton can be run to reproduce the trajectory of uninfected cells, infected cells of lymphocytes T of $CD4+$, free virus particles and specific antibodies CTL (cytotoxic T lymphocyte).

6.4 Computational graphical interface

We have used *Matlab 7.0* to build our computational graphical interface for the Blood-Tor system (see its initial interface in Fig. 21). To set up your own simulation, the following two

Fig. 21. Computational Graphical Interface of Blood-Tor.

parameters need to be chosen: 1. medication potency; 2. treatment adhesion. Furthermore, at the end of each simulation, the graphs of the uninfected cells of $CD4+$, infected cells of $CD4+$, free virus and virus specific antibodies as a function of time are plotted. If the user press the Three Simulations button, the simulation is carried out using the data (treatment adhesion and medication potency) of three HIV positive individuals shown in Table 6.

7. Conclusion

The interaction of multidisciplinary areas is very important to construct and strength biological mathematical models. For instance, the interaction of mathematical modeling and clinical research was crucial in understanding essential features of the HIV infection dynamics. Identifying that HIV is a dynamic disease which encompass different time scales (hours, days, weeks, months and years) was a conclusion that resulted of mathematical modeling combined with perturbation experiments. It was then to identify that these time scales correspond

to important biological processes underlying HIV infection. This knowledge is associated for instance with the recommendation of changing the treatment from monotherapy to combination antiretroviral. After that, HIV has become a treatable chronic disease, with HIV mortality rates approaching those of the general population. Another important practical message from modeling is the necessity that patients continue with the antiretroviral treatment for a period of at least 2-3 years after virus is no longer detectable in blood (Perelson & Nelson, 1999; Yazdanpanah, 2009). In this context, CA compared to differential equations approaches are better choices for modeling HIV infection since they can deal with the variety of observed time scales and also can incorporate the heterogeneity of populations and the local interactions (Sloot et al., 2002). It also important to address questions concerning the sensitivity to parameters raised for instance by Strain & Levine (2002) on the Zorzenon dos Santos & Coutinho (2001) CA model. Burkhead et al. (2009) presented rigorous mathematical results about the time scales and other dynamical aspects of the last model as well as discussed parameter and model changes and their consequences. They gave explanations for the timing in the model supported by numerical observations. The presented results show that fuzzy set theory is a powerful tool to deal with the uncertain, imprecise nature of the virus dynamics. Besides the discussion on mathematical aspects of the models, it is also important to be aware of questions posed by recent studies on HIV and AIDS. Yazdanpanah (2009) discussed the challenges posed by new antiretroviral agents for the management of treatment-experienced patients. They point out it is important to know how to optimize the pairing and sequencing of recently available antiretroviral agents in order to further improve long-term treatment efficacy in patients with multidrug-resistant HIV infection.
In further researches, a cellular automaton that represents initially the blood stream of an individual HIV positive without treatment and afterwards the same individual with treatment would be our main objective.

8. Acknowledgments

The first author acknowledges CNPq, the Brazilian National Research Council, for grant 477918/2010-7.

9. References

Banks, E. (1971). *Information processing and transmission in cellular automata*, PhD of Philosophy, Massachusetts Institute of Technology.

Barros, L., Leite, M. & Bassanezi, R. (2003). The SI epidemiological models with a fuzzy transmission parameter, *Computers and Mathematics with Applications* 45: 1619–1628.

Brazil (2008). Recomendações para terapia anti-retroviral em adultos e adolescentes infectados pelo HIV 2007/2008, BVS – Ministry of Health (in Portuguese).

Bulmer, M. G. (1974). A statistical analysis of the 10-year cycle in Canada, *Journal of Animal Ecology* 43(3): 701–718.

Burkhead, E., Hawkins, J. & Molinek, D. (2009). A dynamical study of a cellular automata model of the spread of HIV in a lymph node, *Bulletin of Mathematical Biology* 71: 25–74.

Caetano, M. & Yoneyama, T. (1999). A comparative evaluation of open loop and closed loop drug administration strategies in the treatment of AIDS, *An. Acad. Bras. Cienc.* 71: 589–97.

Coutinho, F., Lopez, L., Burattini, M. & Massad, E. (2001). Modelling the natural history of HIV infection in individuals and its epidemiological implications, *Bulletin of Mathematical Biology* 63: 1041–1062.

Czaran, T. (1998). *Spatiotemporal Models of Population and Community Dynamics*, Population and Community Biology Series, Chapman & Hall, London, GB.

Dewdney, A. K. (1984). Sharks and fish wage an ecological war on the toroidal planet Wa-Tor, *Scientific American* 251(6): 14–20.

Durrett, R. & Levin, S. (2000). Lessons on pattern formation from planet WATOR, *J. Theor. Biol.* 205(2): 201–14.

Edelstein-Keshet, L. (1988). *Mathematical Models in Biology*, Birkhauser Mathematics Series, McGraw-Hill, New York, USA.

Elton, C. & Nicholson, M. (1942). The ten-year cycle in numbers of the lynx in Canada, *Journal of Animal Ecology* 11(2): 215–244.

Elton, C. S. (1924). Periodic Fluctuations in the Numbers of Animals: Their Causes and Effects, *J. Exp. Biol.* 2(1): 119–163.

Ermentrout, G. B. & Edelstein-Keshet, L. (1993). Cellular automata approaches to biological modeling, *Journal of Theoretical Biology* 160(1): 97 – 133.

Figueiredo, P., Coutinho, S. & Zorzenon dos Santos, R. (2008). Robustness of a cellular automata model for the HIV infection, *Physica A: Statistical Mechanics and its Applications* 387(26): 6545–6552.

Filter, R. A., Xia, X. & Gray, C. (2005). Dynamic HIV/AIDS parameter estimation with application to a vaccine readiness study in southern Africa, *IEEE Transactions on Biomedical Engineering* 52(5): 784–791.

Gilpin, M. E. (1973). Do hares eat lynx?, *The American Naturalist* 107(957): 727–730.

Guedj, J., Thiébaut, R. & Commenges, D. (2007). Pratical identifiability of HIV dynamics models, *Bulletin of Mathematical Biology* 69(8): 2493–2513.

Haase, A. T. (1999). Population biology of HIV-1 infection: Viral and CD4+ T cell demographics and dynamics in lymphatic tissues, *Annual Review of Immunology* 17(1): 625–656.

Hazenberg, M. D., Hamann, D., Schuitemaker, H. & Miedema, F. (2000). T cell depletion in HIV-1 infection: how CD4+ T cells go out of stock, *Nat. Immunol.* 1(4): 285–289.

Hewitt, C. G. (1921). *The conservation of the wild life of Canada*, Charles Scribner's Sons, New York, USA.

Jafelice, R. M., Barros, L. C., Bassanezi, R. C. & Gomide, F. (2004). Fuzzy modeling in symptomatic HIV virus infected population, *Bulletin of Mathematical Biology* 66(6): 1597 – 1620.

Jafelice, R. M., Barros, L. C., Bassanezi, R. C. & Gomide, F. (2005). Methodology to determine the evolution of asymptomatic HIV population using fuzzy set theory, *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.* 13(1): 39–58.

Jafelice, R. M., Bechara, B., Barros, L. C., Bassanezi, R. C. & Gomide, F. (2009). Cellular automata with fuzzy parameters in microscopic study of positive HIV individuals, *Mathematical and Computer Modelling* 50(1-2): 32 – 44.

Jafelice, R. M. & Silva, P. N. (2001). Simulação de presa-predador no planeta Wa-Tor, *In: Congresso Latino Americano de Biomatemática*, Campinas, Brazil (in Portuguese).

Krebs, C. J., Boonstra, R., Boutin, S. & Sinclair, A. (2001). What drives the 10-year cycle of snowshoe hares?, *BioScience* 51(1): 25 – 35.

Krivan, V. & Colombo, G. (1998). A non-stochastic approach for modeling uncertainty in population dynamics, *Bulletin of Mathematical Biology* 60: 721–751.

Lee, Y., Kouvroukoglou, S., McIntire, L. & Zygourakis, K. (1995). A cellular automaton model for the proliferation of migrating contact-inhibited cells, *Biophysical Journal* 69(4): 1284–1298.

Mamdani, E. H. (1976). Advances in the linguistic synthesis of fuzzy controllers, *International Journal of Man-Machine Studies* 8(6): 669–678.

Mamdani, E. H. & Assilian, S. (1999). An experiment in linguistic synthesis with a fuzzy logic controller, *Int. J. Hum.-Comput. Stud.* 51(2): 135–147.

Mielke, A. & Pandey, R. (1998). A computer simulation study of cell population in a fuzzy interaction model for mutating HIV, *Physica A* 251: 430–438(9).

Murray, J. (1990). *Mathematical biology: I. An introduction*, Springer.

Nowak, M. A. & Bangham, C. R. M. (1996). Population Dynamics of Immune Responses to Persistent Viruses, *Science* 272(5258): 74–79.

Nowak, M. A. (1999). The mathematical biology of human infections. *Conservation Ecology* 3(2): 12.

Ortega, N., Barros, L. & Massad, E. (2003). Fuzzy gradual rules in epidemiology, *Kybernetes: The International Journal of Systems and Cybernetics* 32(4): 460–477.

Ouattara, D., Mhawej, M. & Moog, C. (2008). Clinical tests of therapeutical failures based on mathematical modeling of the HIV infection, *Special Issue on Systems Biology* pp. 230–241.

Pedrycz, W. & Gomide, F. (1988). *An introduction to Fuzzy Sets: Analysis and Design*, Massachusetts Institute of Technology, Cambridge, USA.

Pekalski, A. (2004). A short guide to predator–prey lattice models, *Computing in Science and Engineering* 6: 62–66.

Perelson, A. S. & Nelson, P. W. (1999). Mathematical analysis of HIV-1 dynamics in vivo, *SIAM Review* 41(1): 3–44.

Renning, C. (1999-2000). Collective behaviour: Emergent dynamics in populations of interacting agents, *Seminar Artificial Life*.

Saag, M. (1995). *Diagnóstico Laboratorial da AIDS presente e futuro*, Revinter, chapter 3, pp. 27–43. in Tratamento Clínico da AIDS, M.A. Sande and P.A. Volberding, Editors (in Portuguese).

Saila, S. B. (2009). Ecosystem models of fishing effects: Present status and a suggested future paradigm, *in* D. L. G. Noakes, R. J. Beamish & B. J. Rothschild (eds), *The Future of Fisheries Science in North America*, Vol. 31, Springer Netherlands, pp. 245–253.

Shi, V., Tridane, A. & Kuang, Y. (2008). A viral load-based cellular automata approach to modeling HIV dynamics and drug treatment, *Journal of Theoretical Biology* 253(1): 24 – 35.

Silva, C. & Jafelice, R. M. (2010). Estudos de modelos microscópicos do HIV com retardo baseados em autômatos celulares, *Technical report*, Federal University of Uberlândia, Brazil (in Portuguese).

Sloot, P., Chen, F. & Boucher, C. (2002). Cellular automata model of drug therapy for HIV infection, *in* S. Bandini, B. Chopard & M. Tomassini (eds), *Cellular Automata*, Vol. 2493 of *Lecture Notes in Computer Science*, Springer Berlin/Heidelberg, pp. 282–293.

Stenseth, N. C., Chan, K.-S., Tong, H., Boonstra, R., Boutin, S., Krebs, C. J., Post, E., O'Donoghue, M., Yoccoz, N. G., Forchhammer, M. C. & Hurrell, J. W. (1999).

Common Dynamic Structure of Canada Lynx Populations Within Three Climatic Regions, *Science* 285(5430): 1071–1073.

Stenseth, N. C., Falck, W., Chan, K.-S., Bjørnstad, O. N., O'Donoghue, M., Tong, H., Boonstra, R., Boutin, S., Krebs, C. J. & Yoccoz, N. G. (1998). From patterns to processes: Phase and density dependencies in the Canadian lynx cycle, *Proceedings of the National Academy of Sciences of the United States of America* 95(26): 15430–15435.

Strain, M. C. & Levine, H. (2002). Comment on "dynamics of HIV infection: A cellular automata approach", *Phys. Rev. Lett.* 89(21): 219805.

Ueda, H., Iwaya, Y., Abe, T. & Kinoshita, T. (2006). A cellular automata model considering diversity associated with HIV infection, *Artificial Life and Robotics* 10: 73–76.

Wangersky, P. J. (1978). Lotka-volterra population models, *Annual Review of Ecology and Systematics* 9(1): 189–218.

Yazdanpanah, Y. (2009). Multidrug resistance: a clinical approach, *Current Opinion in HIV & AIDS* 4(6): 499–506.

Ying, H., Lin, F., MacArthur, R., Cohn, J., Barth-Jones, D., Ye, H. & Crane, L. (2007). A self-learning fuzzy discrete event system for HIV/AIDS treatment regimen selection, *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on* 37(4): 966 –979.

Zadeh, L. A. (1965). Fuzzy sets, *Information and Control* 8(3): 338–353.

Zhang, Z., Tao, Y. & Li, Z. (2007). Factors affecting hare-lynx dynamics in the classic time series of the Hudson Bay company, Canada, *Climate Research* 34(2): 83–89.

Zorzenon dos Santos, R. M. & Coutinho, S. (2001). Dynamics of HIV infection: A cellular automata approach, *Phys. Rev. Lett.* 87(16): 168102.

7

CA in Urban Systems and Ecology: From Individual Behaviour to Transport Equations and Population Dynamics

José Luis Puliafito
Universidad de Mendoza
Argentina

1. Introduction

One way of seeing Cellular Automata (CA) is as cell- based computational models for describing the evolution of spatially distributed systems. Each cell represents a "local" state of the system that can vary according to its past states and to the present states of a "vicinity" of cells, following some set of relations known as "transitions rules".
More important than how these transition rules are (i.e linear, non linear, discrete, etc,), is that distant parts of such system can interact one to another only through its neighbours; in other words, what we are actually considering in CA models, is that the system obeys *the principle of locality*. For this seems to be the case of most systems in nature, CA models have found potential applicability in a wide variety of phenomena, ranging from macroscopic scales, like urban systems, down to microscopic scales like in solid state physics. J.F Nystrom (2001) has even argued in favour of the idea that fundamental laws of physics should arise from simple transitions rules of some Universal CA, in a structured space following R. Buckminster Fuller's synergetic geometry.
This brings us to a central point, which is that, in nature, space is as essential as time for describing any process; disregarding if we are more interested in watching at the temporal behaviour of certain group of state variables or if we are more interested in taking static pictures of some distributed properties in space, there will be always a spatiotemporal evolution process taking place behind.
A good example are urban and environmental systems; social scientists have been discussing since long ago how population and economy of regions interact and evolve through the years, while geographers and urban analysts have been doing it looking at its spatial structures. Both have contributed in equal parts to our present understanding of sustainable development. However, ¿can economists explain development without considering where was located the infrastructure support? or ¿can urbanists explain the structure of a city without considering the historical circumstances? Both views tend to describe one aspect of the evolution looking at the other as frame constrains, usually given in terms of literal stories. The same happens in many other fields of science treating with complexity.
A more modern view stands on the growing availability of informatics tools, and pushes towards constructing spatiotemporal models. But this is not a simple task; most attempts

flow amongst "top-down" approaches, based on continuity equations in partial derivates - including fractional order diffusion equations for explaining behaviours with long-range dependency (Angulo J.M, et al. 2001)- and "bottom-up" approaches, mostly given by discrete rule-based CAs (Park S. and Wagner D. F., 1997). While the former go more for the classic type, which puts emphasis in an extensive view of the system (that is in its behaviour and consistency as a whole), the latter are more of the evolutionary type, giving more emphasis to a detailed view, trying to describe self-organization and innovation proper of complexity.

However, as discrete space-time models have become more attractive, due to the intensive use of dynamical raster GIS (Geographic Information Systems) (Batty, M, 1996; Mitas L. 1997 et al.; Park S. and Wagner D. F. , 1997), different kinds of CAs , as well as seamless discrete-continuous approaches, are opening new theoretical avenues.

For instance, as regards mobile agent-based CAs, the number of agents (population) grows initially from implanted "seeds" reproducing and spreading on the back cells, in accordance with transition rules and information on their development capability held in different GIS layers (Batty M. and Torrens P., 2001).

Not far from these, some seamless continuous/discrete approaches face the modelling problem in terms of a particle-field duality, just as in the path sampling method used in physics for solving continuity equations (Mitasova, H. and Mitas L., 2000). Multidimensional complexity can be treated herein by means of particles and fields in different scales. Likewise, some approaches use spatiotemporal convolution equations with kernels limited in space (i.e gaussian or similar), or even space-variant kernels (heterogeneous) (Wikle C., 2001), in a way that complex spatiotemporal processes are described as the propagation of dispersive or non-dispersive wave packets.

The distinctive feature of the seamless and mobile agent-based CAs models is that they use particles - or rather pseudo-particles - as an attempt to match the continuum response of an extensive view with the discrete and evolutionary behaviour of a detailed view; these can be considered as descriptions halfway between classical physics kinetics and unstable system dynamics. Issues to be primarily considered herewith are: a reduction in the amount of information involved, the interlacing of layers or embedding of contributing models, as well as the setting of scales for representative particles of the included processes.

The modelling of ecological systems offers also good examples; the emergency of complex spatiotemporal patterns in the population dynamics of certain species has been since long time of great interest in Ecology. Random walking and diffusion equations are used to describe the movement of animals in their own environment, and to forecast their spatial distribution under the influence of the diverse territorial heterogeneities (Jeanson R. et al., 2003). Such models are found on a regular basis but there is still a long conceptual way to go.

Complex spatiotemporal patterns in the activities carried out by some social insects, such as ants and termites, reveal that individuals can collectively do better at performing tasks than isolated. This is not only observed in the typical pattern scales, usually far larger than the size of individuals, but also in their shape, featuring arrangements in various delicate and regular structures. Despite individual randomness and limitations, collective structures arise effectively in response to several functional and adaptive requirements (protection against predators, the substrate of social life and reproduction, thermal regulation, etc.) (Theraulaz G. et al., 2003).

Twenty years of research have revealed that the origin of hierarchical complexity is more a consequence of the multiplicity of individual responses to stimuli, derived of relatively

simple behaviours, than of the ability of each insect to process a large amount of information. Hence, the resulting patterns seem to emerge from non-linear interactions among individuals and between individuals and their environment, all this through mechanisms like templates, stigmergy and self-organization (Theraulaz G. et al., 2003; Ball P., 1998).

These features in particular have pushed traditional temporal dynamic analysis towards incorporating more explicitly space (Spatial Ecology), through metapopulation and transition rules models like the Cellular Automata. The interest is in the link between the spatial structure of the environment and of the occupying population with the species features, their development, survival and even their diversity [Pascala, S. and Levin, S. 1997; Tilman,D and Kareiva, P. 1997].

The latter also points at phenomenological models with differential equations in partial derivatives, such as the reaction-diffusion equations based on the Alan Turing model (1952). This was originally applied to the morphogenesis of skin spots in animals like zebras, jaguars and leopards, and later extended, by several authors, to nearly all the range of biological and ecological patterns, being cellular morphogenesis and the spatial segregation of species included. Basically, it describes the non-linear interaction of two- species concentrations: one is an "activator" (rather of local action) and the other an "inhibitor" (of a longer reach), so periodical structures rise as a consequence of different diffusion speeds (Meinhardt, H., 1982). An outstanding example in the biological level is the chemotaxonomic spatiotemporal behaviour of two bacterial species, which can be externally controlled and shapes propagating waves and patterns (Lebiedz D. and Brandt-Pollmann U., 2003).

The variety of approaches is not as much a consequence of the type of system under consideration, as of the need to integrate multidimensional interactions at various levels, where a spatiotemporal model rises from any of the following (Popov V.L. and Psakhie S.G., 2001):

a. the macroscopic dynamics of the system and by finding solutions to partial integro-differential equations (if known);
b. the microscopic dynamics of the real system and by finding interaction laws through molecular dynamics methods or first-principle methods;
c. the replacement of the real system by a certain medium model (having rougher microscopic behaviour but the same macroscopic dynamics as the former), while formulating proper transition laws.

The third type of approach is where CA and seamless models are actually placed; in particular the use of Cellular Automata has widely spread because of its intrinsic capacity to simulate complexity, specifically self-organization and innovation. However, and going back to the beginning, it should be bared in mind that such models are eventually tensorial computational methods based on finite spatial cells, thus defining an *excitable elastoplastic medium* that represents the species-space/environment system in question. In any case, Cellular Automata can successfully model several types of excitable media, not only due to some insensibility of "macroscopic" dynamics in relation to the structure and nature of interactions in their "microscopic" order, but also to the fact that most systems and even hypothetical mathematical objects, are described by some kind of *transport equations* (Popov V.L. and Psakhie S.G., 2001).

It can be surprising that despite the obvious conceptual division between the animate and the inanimate worlds certain population phenomena are described similarly. In fact,

reaction-diffusion systems are frequently found and rather important in many areas of physics. For instance, through the *band theory*, crystalline solids such as semimetals and semiconductors can be described on an electrical basis by means of two charge transport equations, one on electrons (negative charge) and other on holes (positive charge) in specific energy bands. The concentration of each carrier can be described in a similar way to Turing's, since both transport equations are coupled through the generation/recombination of carriers, similar to predator-prey interactions in Ecology.
As the active and inhibitor species rise naturally herewith, leading to stigmergy-like based mechanisms, *¿is it possible that some of the methods and principles used in solid state physics be also applicable to ecological and urban systems?*; if so, and even though living systems would have more plasticity, a crystallographic metaphor would be useful to model certain aspects relevant to spatiotemporal evolution in social species, at least under stationary or quasi-stationary conditions. This approximation has been studied and applied to spatiotemporal modelling of urban areas, thus showing its viability and potentiality at explaining several heterogeneously distributed urban phenomena (Puliafito, J.L. 2006).
We must bear in mind herewith that at describing the spatiotemporal dynamic evolution of populations of real individuals through transport equations, one is not only implicitly considering the existence of definite interactions of the species with its space/environment, but also stability regions in the associated state space that are similar to the energy bands in solid materials (multistability; Theraulaz G. et al., 2003). Therefore, either systems can stray away slightly from the previous dynamic relations so that restoring forces will tend to preserve evolution within a states region (linearity, elasticity), or can stray away largely with transitions among regions (non-linearity, plasticity). In this sense, experience proves that social behaviour and complex and regular spatiotemporal structures usually emerge under conditions where species reach some critical spatial density.
In such train of thoughts, herein an ecosystem is not the mere association of interactions in terms of the whole, or the sum of strongly-interacting independent elements, but a rather coherent sum of elementary units made up of living individuals and their immediate surrounding space-environment; the latter being regarded as a multidimensional representation of the resources needed for its survival, physical space in itself included.
A review of some of the investigations done by the author dealing with the above questions is presented in this chapter.

2. A bridge from the stochastic behaviour at the individual level to the associated behaviour at the collective level

A research program dealing with all these features should start at modelling generic individuals as automata exploring the environment, capturing and feeding from discrete units of matter and energy, thus developing some sort of random-walk mostly confined to a certain territory. This leads, as it will be briefly shown here, to spatiotemporal behaviours that resemble quantum stochastic systems. Apart from the theoretical interest, it yields the possible application of simplifying analogies to the population dynamics of dense concentrations –even in a restrictive manner -, as it lays a bridge towards a collective description similar to the band theory of solids (Puliafito, José L and Puliafito, S. E. 2006).

2.1 First class Bioautomaton – Langevin equation

Let us consider some type of autonomous homeostatic device, which, for our purposes, we can call *bioautomaton*. In an elementary class (first class bioautomaton), such an ideal device

is an open biophysical system moving step-wise and randomly in space, aiming at capturing, storing and processing discreet units of matter/energy (resources), thus assuring its "survival". It can be seen as a black box excited by Poisson impulses and responding with limited spatial displacements through an appropriate transfer function. Due to these mechanical displacements, the resources that are available in the near-by space–environment can be captured. The latter can also be considered as a black box excited by a random step function, giving such energy impulses to the bioautomaton. (Fig 1)

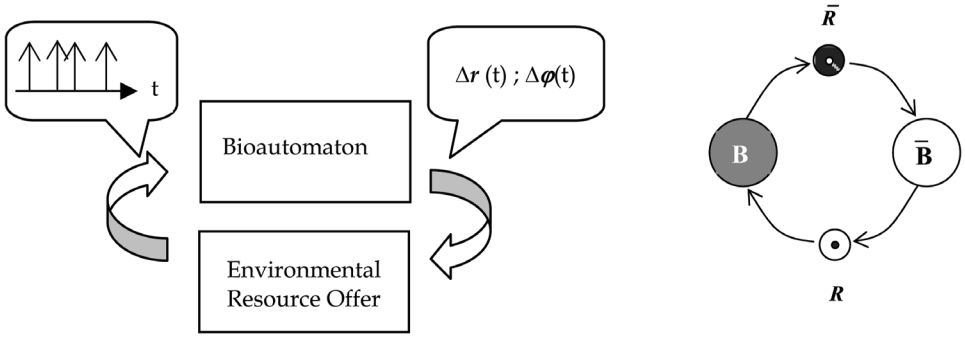


Fig. 1. Left: Scheme of the bioautomaton/environment interaction; the bioautomaton has an internal reserve of energy . Heat exchange fluxes are not displayed for simplicity. Right: The bioautomaton-medium interaction seen as a closed system; B particle is the bioautomaton and $\bar{B}$ antiparticle represents the near-by medium; the exchange is given by a discrete flux of resources R and of residues $\bar{R}$.

The evolution in the state space of the whole system (bioautomaton / environment) is a stochastic process, depending, on one side, on the efficiency of the bioautomaton to collect resources and to adequately use its internal energy reserve and, on the other side, on the environmental offer and its renewal capacity. There will be stationary or quasi-stationary random solutions, as long as the expected value of the rate of energy consumption per period between impulses is higher or equal to the average minimum consumption rate:

$$< \boldsymbol{\varepsilon_{\delta i} / T_{\delta i}} >= \varepsilon_0 / T_0 \geq (\delta\varepsilon/\delta\tau)_{min} \quad (1)$$

Eq. (1) can be considered as the *first-class functional* of a bioautomaton or "survival" functional, where $(\delta\varepsilon/\delta\tau)_{min}$ plays a similar role to basal metabolism in living organisms. Here, the device's "survival" consists of a set of conditions resulting in the sustenance in time of its internal reactions, within a relatively steady range, balancing dynamically the energetic exchange with the environment.

Since under stationary conditions the bioautomaton's movement and survival are limited to an optimal use of its internal energy reservoir, a certain potential function can be associated to this storage, as a measure of the probability to capture new resources. This can be defined as a spatiotemporal convolution between a certain window *S(r)*, representative of the perception and capture radio of the bioautomaton, and the spatial density of resources ρ(*r*) (ζ it's a process constant):

$$U_{br}(\boldsymbol{r}, t) = -(1/2\pi\zeta) \int_{\tau} S(\boldsymbol{r}(t) - r') \rho(r') \, d\tau \quad (2)$$

When taking a gaussian window and a localized distribution of resources (eg. disc type) a "well" spatial function is obtained, which recognizes approximately the regions given by the three degrees of homogeneity in classical mechanics (K_h= 2 parabolic for $r \leq 1{,}5\ r_0$, K_h=1 linear for $1{,}5\ r_0 < r < 2r_0$ and K_h= -1 newtonian for $r \geq 2\ r_0$, with r_0 as a characteristic radius). Unlike a classical potential, which is determined by the medium, U_{br} depends on what the environment can offer as regards means, as well as the degree of utilisation (or efficiency) the bioautomaton can get out of them; that is, it represents *the expected interaction bioautomaton-environment*. Thus, its interpretation as a potential function is conditioned to the resulting movement being a stationary or quasi-stationary process, or, in other words, being an efficient estimator of the spatial distribution of resources.
A generalized *Langevin stochastic differential equation* derives from the previous definitions for the bioautomaton, which can be analysed from partial solutions for the homogeneity regions above given (3).

$$m.\frac{d^2}{dt^2}\,\bar{\mathbf{r}} + f\,\frac{d}{dt}\,\bar{\mathbf{r}} - \frac{\partial}{\partial \boldsymbol{r}} U_{br}\,(\,\bar{\mathbf{r}}, t\,) = \bar{\mathbf{F}}_{\mathbf{ex}}(t) = m.\ \bar{\boldsymbol{n}}(t) \qquad (3)$$

Note that equation (3) has reduced the quite complex interactions to the dissipative stochastic movement of a m mass and f friction punctual particle, subjected to certain excitation and restitution forces dependent on the $U_{br}(\,\boldsymbol{r}, t\,)$ virtual potential. Formally, it can be interpreted as a generalized type of Brownian movement, where $\bar{\boldsymbol{n}}(t)$ represents white shot noise.

2.2 Behaviour in K_h = 2 zones

Near the distribution centre of $\rho(r)$, $U_{br}(r)$ takes the shape of a second degree parabola (K_h = 2), in a way that the potential gradient (the potential reactive force) is approximately proportional to displacements:

$$\frac{d^2}{dt^2}\,\bar{\boldsymbol{r}} + \beta\,\frac{d}{dt}\,\bar{\boldsymbol{r}} + \omega_0^2\,\bar{\boldsymbol{r}} = \bar{\boldsymbol{n}}(t) \qquad (4)$$

with $\beta = f/m$ and $\omega_0^2 = k/m$, which corresponds to the movement of a particle in a viscose medium under the action of a central field. In the case of the bioautomaton, β must be understood more generically as the relation between the total dissipative forces (outer and inner) and the total equivalent mass that includes the inert mass and the associated biomass. The bidimensional problem can be described in terms of an analytical process with a complex random variable $\boldsymbol{r}(t) = \boldsymbol{x}(t) + j\ \boldsymbol{y}(t)$. The stochastic processes $\boldsymbol{x}(t)$ and $\boldsymbol{y}(t)$ are also described through independent differential equations of the type given in (4), coupled through proper coefficients $\omega_{0x} = \omega_{0y} = \omega_0$ and $\beta_x = \beta_y = \beta$, which presupposes spatial isotropy.
The essential properties of the movement originate in the characteristic equation for the autocorrelation of any of the two components. The weakly dumped harmonic case is of particular interest $(\beta/2\omega_0 < 1)$, as it describes a range of solutions corresponding to a *stochastic oscillator* in which trajectories are stochastic "orbits":

$$\boldsymbol{r}(t) = \mathbf{r}\,.\,\exp\,[-0{,}5\,\beta t + j\,(\,\omega_d\,t + \boldsymbol{\varphi}\,)] \qquad (5)$$

with $\omega_d = \omega_0\,(1 - \beta^2/4\omega_0^2)^{1/2}$; $\mathbf{r} = (\mathbf{x}^2+\mathbf{y}^2)^{1/2}$; $\boldsymbol{\varphi}$=atan $(\mathbf{y}/\mathbf{x})$. This case gives the longer possible *average life times* of the bioautomaton ($\tau \sim 2/\beta$), thus becoming the most appropriate for the definitions above given.

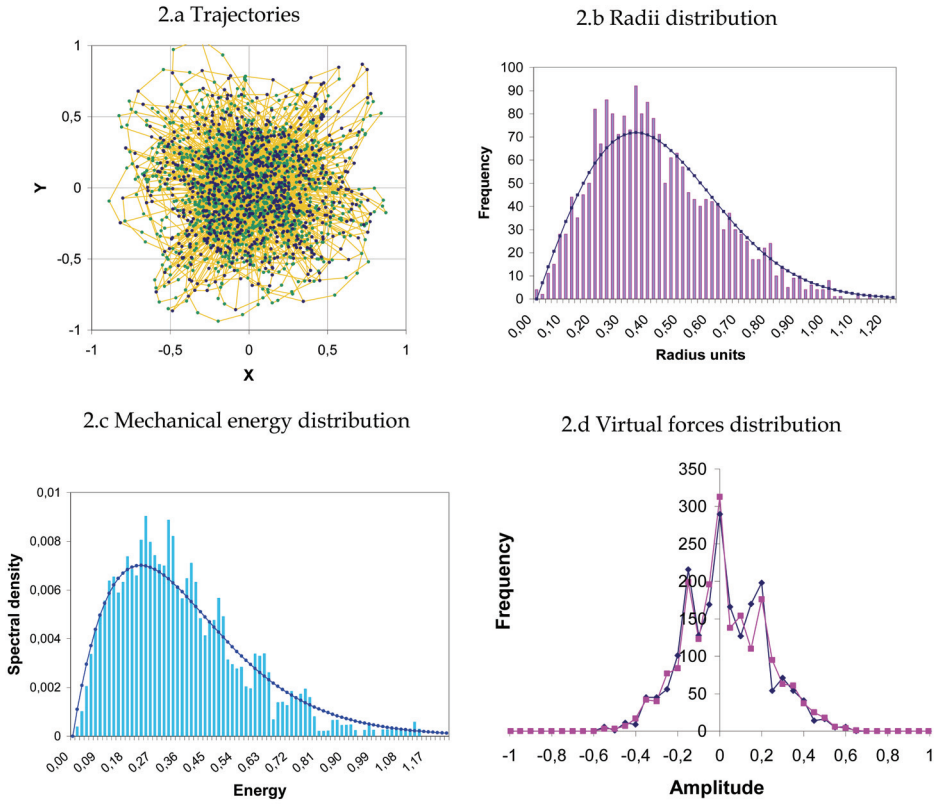


Fig. 2. Numerical simulation of bioautomaton in K_h= 2 zone, with Poisson impulses excitation and random directions; simulation parameters are: Mass m=10; Friction f=3.145; Elastic constant k= 0.987 (ω_0 = 0,31416; β = 0,3145; ξ= 0,500)

Considering in particular when **n**(t) is shot noise, representing discrete supply events, the resultant of the apparent excitation forces $\bar{\mathbf{F}}_{ex}$, can be expressed as follows:

$$\bar{\mathbf{F}}_{ex}(s) \approx \sum_i \gamma.\Delta s.\delta(s-s_i)\ \bar{s}^{\circ} \rightarrow \bar{\mathbf{F}}_{ex}(t) = \sum_i (\varepsilon_i / \bar{v}_i)\ \delta(t - t_i) \qquad (6)$$

In the first expression of eq. (6) γ is the apparent density of energy per longitude unit, Δs the mean step and $\delta(s-s_i)$ the delta Dirac function for $s=s_i$, thus describing an impulse train with events located in (i) random positions over the s trajectory. Random positions s_i can be considered as independent events, resulting in a Poisson process with a density parameter α_s =N/<s> of points, determined by the distribution of resources in space, and a expected value $\langle \mathbf{F}_{ex} \rangle = \gamma\ \delta s\ \alpha_s$. The second expression of eq. (6) is in explicit function of time, where $\bar{v}_i$ is the instantaneous vector velocity of the bioautomaton over trajectory s and ε_i the specific energy of the resources captured in s = s_i random positions. In this way average trajectories in stationary processes will *depend directly on the number N of captured resources* . In fig.2 a numerical simulation of the stochastic differential equation (4) is shown. Figure (2.a) shows the trajectories and the encounter positions with resources corresponding to the

input impulse sequence. Hence, *a cloud of excitation points* is associated to the trajectories, which is denser in the centre and dilutes to the outside. As expected, the position radii follow a Rayleigh distribution (fig 2.b); accordingly, the distributions of x-y distances, as well (as of x-y velocity components), follow a gaussian form. The power spectrum of radii has also a concordant distribution in frequency, given by a second-order transfer function shifted by the natural angular frequency of the system.

Some of the peculiar properties of shot noise excitation arise already when the distributions of the mechanical energy of the system and of the virtual forces are considered. The distribution of the mechanical energy (fig 2.c) (as in addition to its transitions) has a spectral density that nearly follows a Planck type distribution with two degrees of freedom (the correlation factor is approximately 0.97):

$$F(\varepsilon) = A.\ \varepsilon^2 / \exp\left(\varepsilon/\varepsilon_T - 1\right) \tag{7}$$

where ε represents energy, ε_T a "thermal" equilibrium energy , and A a proper characteristic constant. The third factor is the Bose-Einstein distribution, but here ε is squared instead of at a cubic power as in the Planck radiation law. In this sense, while the distribution of the x-y instant components of the input virtual forces follow a sine law (vector decomposition in uniformly distributed random phases), their mobile media (within a window as long as the feeding period T*) show the expected tendency to gaussian distributions, but with a clear fine reticular structure (fig. 2.d).

The former results show the need to consider the *quantification* of the bioautomaton's behaviour as regards the *number of capture events associated to the noise density* (N). In fact, it is possible to analyse the device behaviour by decomposing its excitation in terms of *a random sum of k- input modes*, where k=1 always represents one input impulse per feeding period T*, k=2 represents a random sequence of just two impulses per period, k=3 represents three impulses, and so on. Denoting $\check{z}(t)$ as the mobile average of excitation forces $z(t)$, one gets the distribution $f_{\check{z}}(\mathbf{z})$, shown in figure (2.d), by doing

$$f_{\check{z}}(\mathbf{z}) = (1/\pi)\ \sum_k\ b_k f_{zk}(\mathbf{z}) \tag{8.a}$$

where: $$f_{zk}(\mathbf{z}) = \int_0^{\Omega_{max}} J_0{}^k(a\,\Omega).\cos \Omega z\, d\Omega\ ;\ b_k\ =\ (\lambda.T^*)^k\,.\,\exp(-\lambda.T^*)\ /\ k! \tag{8.b}$$

In eq. (8.a) and (8.b) $f_{zk}(\mathbf{z})$ is the *k-modal component* of the distribution density, expressed as the cosine Fourier transform of the k power of the first kind Bessel function of zero order, in the sthocastic frequency domain Ω; Ω_{max} is a cut-off stochastic frequency which rises from the forces reticular structure in stationary conditions. At the same time b_k is a weight factor of mode k, expressed as a *Poisson probability coefficient*.

A k=0 mode can be also defined, which means that no impulse is arriving (no resource is captured), so that the movement is carried out just by the use of the internal storage of energy. According to the *virial theorem* (case K_h= 2), the total average mechanical energy is *half of the system total energy in the k=1 mode*.

Taking into account the previous considerations it is also possible to decompose in k-modes the average value of the angular moment, as well as of the mechanical energy. In fact, the expected angular moment when excited by N capture events per feeding period is:

$$\langle \mathbf{M}_\phi \rangle^2 = N^2\,\varepsilon_0{}^2 / \omega_0{}^2 = m_0{}^2 \omega_0{}^2 r_0{}^4 \tag{9}$$

In eq. (8) one can immediately define an *action constant* $\varepsilon_0/\omega_0 = a/2\pi = \check{a}$, which represents the average rate of the energy consumed per capture event, associated to *a certain consumption capacity* of the bioautomaton. In this way, considering k=0 as the *basal mode*, k-modal components in K_h=2 zones for the angular moment and the energy can be written as:

$$M_{\phi 0k} = (k + ½) \check{a} \tag{10}$$

$$E_{0k} = (k + ½) \check{a} \omega_0 \tag{11}$$

which have the same form as in the *quantum harmonic oscillator*. However, one must consider that these are not pure states but the *average values* of *associated state groups*, in a way that their composition, through the Poisson coefficients given in (8.b), define the general *mixed state* of the Bioautomaton in zones K_h=2.

As the internal energy of the bioautomaton, given by kinetic energy added to the storage energy, is $E_i = m_0 \upsilon^2$, where $\upsilon = v_0$ can be taken as a *typical velocity of bioautomaton-medium interaction*, other virial relationships can be drawn from here in terms of *associated wavelengths* as well as stating the relativity of the average exploration radio and the effective mass of the bioautomaton, respect of its actual average velocity *v*. In fact, alterations can occur while the bioautomaton still keeps the stationary regime; if ω is the apparent frequency of the average forced excitation regime, produced by the search movement of the bioautomaton, the relative frequency ω/ω_0 results from the variation of the average relative velocity bioautomaton v/υ. These relationships express the average spatial response of the bioautomaton- medium feedback system when trying to keep its stationary regime under alterations of interaction parameters.

Equations (10) and (11) and their associated virial relationships establish as a whole, an *allometric relation* between the resource flux and the device effective mass, of the type $(\delta\varepsilon/\delta\tau)_{min} \sim a\ m_0{}^b$ similar to the ones observed in the real biological world, according to the *theory of biological similitude* of Max Klieber (1932) and to research works carried out more recently like Hemmingsen (1960) and Günther et al. (1992). [1]

2.3 Behaviour in distant zones

If the device is deployed far from the resource centre, that is, within an interval of distances of the virtual potential corresponding to regions K_h=-1, the generalized Langevin equation has no lineal term on distance but one of the Newtonian type (-1/r). In this case two kinds of behaviours can be basically considered, one that is stationary and another that is a transition from K_h=-1 to K_h=2.

In the first case (K_h=-1) the average trajectories are longer and with less chances of capturing resources. A group of stationary solutions here demands lower frictions or higher energy per resource. Besides, the selective character of the non-lineal form of the differential equation makes stationary solutions critically dependent on the set of values chosen for the device parameters and its excitation. In this case trajectories are also mostly confined into certain average radio, but with spatial distributions that are compounded of various

[1] The magnitude $(\delta\varepsilon/\delta\tau)_{min}$ represents the basal metabolism, m_0 is the mass expressed in kg weight and *a* and *b* are proper allometric parameters. Max Klieber first proposed the allometric relation for most mammals adopting b=0,738, and Hemmingsen and other authors extended such relation even for different homeothermic, poikilothermic and unicellular species with b=0.75.

anisotropies. This is partly due to an additional degree of freedom (stochastic rotation), and partly to the group composition of k-modal solutions similar to the ones seen before.
The second case refers to a bioautomaton having inadequate parameters for keeping a stationary dynamic behaviour in region K_h=-1, but instead they are adequate for region K_h=2. Sinulation tests show that there is a quite fast transition from the first to the second region, passing through region K_h=1. Once the device reaches the parabolic region, its behaviour becomes stationary again, as described before.

2.4 Comparison with quantum-stochastic systems

The former points suggest certain similarity to quantum stochastic systems, mainly due to the discrete character of the resource absorption and that the movement takes the form of a random step sequence, confined more or less to a certain exploration area.
In order to go deeper into this similarity, it is necessary to focus on the dynamics of the bioautomaton-environment system from the possible transitions of states. In this sense, apart from the stationary movements seen above, there can exist forced displacements that would result from the *virtual movement of the resource centre.* This would occur, for instance, when the resource flux diminishes in an originally dense zone. A *drift* or a *migration* of the bioautomaton can be conceived here. In fact, if diminishing the potential storage turns into an estimation of the distance to the resource centre equivalent to a K_h=1 region, slight state changes would force the bioautomaton to "accompany" the virtual displacement of the centre (drift). If diminishing the potential storage becomes so large that the estimated resource centre occurs at a virtual distance equivalent to a K_h=-1 zone instead, a transition would take place (migration).
This can be alternatively appreciated from the *Chapman-Kolmogorov* equation, which is a property of the transition functions in Markov processes. Due to Kolmogorov, *progressive and regressive diffusion equations* can be derived from it, being the regressive the *Fokker-Planck* diffusion equation. As a Markov process (increasing times) is also so in an inverted manner (decreasing times), the progressive equation can be understood as an *antidifussion,* or as the diffusion of trajectories of an *antiparticle*, which would represent the virtual motion of the resource centre. Hence, interaction must be seen as a rather symmetric exchange between two poles; if the position is fixed in the bioautomaton, an incident flux of resources is seen, while if the position is fixed in the resource centre an incident flux of "voids" (or residues) is seen (fig. 1 right).
In the strict stationary case, the progressive and regressive diffusion equations present a *closed symmetry*, thus implying that the consumed resources and the residues produced by the automaton are equalled to the resources produced and residues processed by the environment; in a drift (the bioautomaton follows closely the resource centre) there is a *practically closed symmetry* (quasi-stationary regime), and it is possible to refer such equations to a system of mobile coordinates leading back to the previous case. Finally, symmetry breaks down definitely during a migration and the said equations express two rather independent trajectory fluxes, one for the particle and the other for the antiparticle.
As for what was stated above, the pair of equations generalized for stationary or quasi-stationary bidimensional movements (with means and variances not depending on the absolute position) show somehow the *expected flux of resources and residues* for growing times ($t > t_0$) from the point of view of particle B:

$$\begin{cases} \partial p/\partial t + \bar{\nabla} \bullet (\bar{v}_F\, p) + D\, \nabla^2 p = 0 \\ \partial p/\partial t + \bar{\nabla} \bullet (\bar{v}_R\, p) - D\, \nabla^2 p = 0 \end{cases} \quad (12)$$

with $p = p(x,y,t;x_0,y_0,t_0)$ the bidimensional transition probability, v_R a "regressive" velocity, v_F a "progressive" velocity and D a diffusion coefficient. Following for instance Smolin L., 2007 (inspired in Nelson E., 1966) a wave equation similar *to Schrödinger's equation* can be derived, with $ă^2 = 2m_0^2 D^2$:

$$j\, ă\, \partial\psi / \partial t = - (ă^2/2\, m_0)\, \nabla^2\psi + U_b.\, \psi \quad (13)$$

Two fields are here defined, ψ and its conjugated ψ^*, associated to *the normalized average flux density* of resources and residues, in a way that their product is the transition probability $\psi\psi^* = p\ (x,y,t;x_0,y_0,t_0)$, consequently establishing the quantum similarity of the system bioautomaton-medium[2].

Finally, the k-modal decomposition of ψ and ψ^* can be incorporated, introducing sets of orthogonal wave functions or *associated wave function groups* $\psi = \sum_k B_k\, \psi_k\, (x,y,t;x_0,y_0,t_0)$ and $\psi^* = \sum_k B_k\, \psi_k^*(x,y,t;x_0,y_0,t_0)$; where $B_k = b_k^{½}$ with b_k the Poisson coefficients. They describe the expected configuration of resources and residues by means of its k-modal wave functions: the ψ_k ones associated to the incoming or incident flux and the ψ_k^* associated to the outgoing or reflected flux, thus producing a *mixed general state* of the bioautomaton, as compared to a *quantum stochastic system.*

However, it should be emphasized here that the bioautomaton is not a quantum system but a *classical system with quantum similarity,* which eventually falls near the treatment of quantum dissipative systems given in modern ontological interpretations of Quamtum Mechanics, such as those of *consistent histories,* according to which the purpose of a quantum theory is to predict instances of probabilities of various alternative histories[3]. The consistency criterion states that a system's history can be described on the basis of classical probabilities for each alternative history, compatible (consistent) with Schrödinger equation.

2.5 Transition to collective systems

The bioautomaton can only be considered as a very vague and simplified representation of a biological organism. Notwithstanding, taken as a basic component of a relatively stationary population, and far from describing the life cycle and reproductive function of living beings, still can be used for studying some aspects of real collective behaviours. For that, it is not

---

[2] The average Hamiltonian of the bioautomaton-medium system is H (x ;y) = kinetic energy + resource energy + storage = $½\, m_0 v_0^2 + \xi' m_0 v_0^2 + U_b$; the resource component $H_{res} = \xi' m_0 v_0^2 = m_0 u_0^2$ plays here a similar role to Bhom's *quantum potential* (Bohm, D. , 1952; Smolin L., 2007)

[3] This is confirmed in various elements as in the generalised Langevin equation (3), which is equivalent to the one proposed by Magalinski in 1942, later continued by other authors as a general method to analyse quantum dissipative systems (Hänggi P., Ingold G-L., 2005). Or in the conclusions reached by Wang Q. A. (2005), which states that there exist commuting and uncertainty relations in the classical stochastic processes similar to the ones predicted by Heisenberg, and also by Faigle U., Schoenhuth A. (2006), which establish a general type of stochastic models with quantum prediction (Quantum Predictor Models), out of which the bioautomaton would be a subclass. Likewise, it is sustained in the very derivation of the Schrödinger's associated equation, which although following a formalism similar to Fényes and E. Nelson's (Smolin L. 2005), here it is relatively direct and gives rise naturally to the equivalent of Bohm's quantum potential.

hard to imagine bioautomata that are being subjected to second class functionals, by which mutual interactions can have complementary or exclusion symmetries, or even subject to third class functionals by which group survival can be optimised. Beyond the interest in drawing or not quantum analogies, its importance resides, mainly, in the effects the extension of the previous outcomes have over the behaviour of an ensemble of devices, and eventually over the population dynamics of biological species.
In this sense, the fact that the statistical behaviour of a bioautomaton can be represented in average by means of wave functions, allows to glimpse also stationary or quasi-stationary solutions regarding group behaviour, resulting from the superposition of individual wave functions. For such reason it is quite possible that a bioautomata ensemble tends to a sort of *periodical spatial structure* in *fluctuating cells.* Hence, dynamics can be described in terms of transport equations (arising from eq. 12) and framed within some *appropriate band theory.*
Accordingly, a *basic equivalent model* could be outlined in terms of a virtual substrate with two energy bands: a population band and a resource band where their associated pseudo-particles – *the inhabitant and the recurson* – represent (in principle in an anti-symmetrical way) the interacting population and their space- environment structure correspondingly. Within this context, a high-density ecosystem, can be compared in some sense to an elastoplastic network and thence treated in a similar way to the solid state of matter; that is, as state transitions in a *pseudo-crystalline virtual substrate* subdued to a general exclusion principle.

3. Modelling urban dynamics based on an analogy to solid state physics

Urban regions in particular, being high density human ecosystems, tend to present statistical virial relations and highly structured territorial occupations. The crystallographic picture can bring a new insight to the modelling problem, providing additionally some of the powerful tools used in the solid state theory.
As for the pseudo-particles, here the inhabitant can be mostly conceived as an average individual, while the recurson more like a hamper of resources, depending on the present population needs and the cultural trends; as in quasi stationary frames, changes in the hamper composition and its weights are limited, the total composition can be roughly replaced by a single representative resource, which in most cities can be accomplished by using the available statistics on the real estate values. Complicated interactions can be treated hereby as transitions of the process agents within groups of states or energy bands, ruled by the well known Fermi- Dirac statistics based on the exclusion principle. Hence, the bands structure will represent the organization and hierarchies of the system, and the Fermi energy level a measure of its density.
Beyond this standing point, the analogy also provides a way to introduce a proper potential, by defining an equivalent population charge on the basis of the spatial properties of the interaction between complementary agents. It is possible then to build a static representation of the urban region in terms of the field theory and therefore to represent the spatiotemporal processes by means of coupled transport equations of opposite charge carriers (Puliafito, José L. 2006).

3.1 Characterization of an urban region

A band model and the characterization of pseudo particles can be derived, for a given urban region, observing the statistical properties of the spatial distributions of population and of

real estate values. A case study was developed for Great Mendoza, an urban region of about 850,000 inhabitants (1995) located at the foot of the Andes Mountains, 32.8º South latitude and 68.8º West longitude in the Province of Mendoza, western Argentina.
A statistical assessment developed from a GIS raster representation of Great Mendoza's distributions -using 350 x 350 m2 grid elements and official census data collected between 1990 and 1992-, reveals the existence of two subsystems: one dense and central and the other rather diluted and peripheral (fig 3). The former contributes to the characterization of the main urban agglomerate whereas the latter to its interphases of expansion. The dense subsystem can be represented appropriately by statistics of the Fermi-Dirac type (FD) and its respective spectral densities:

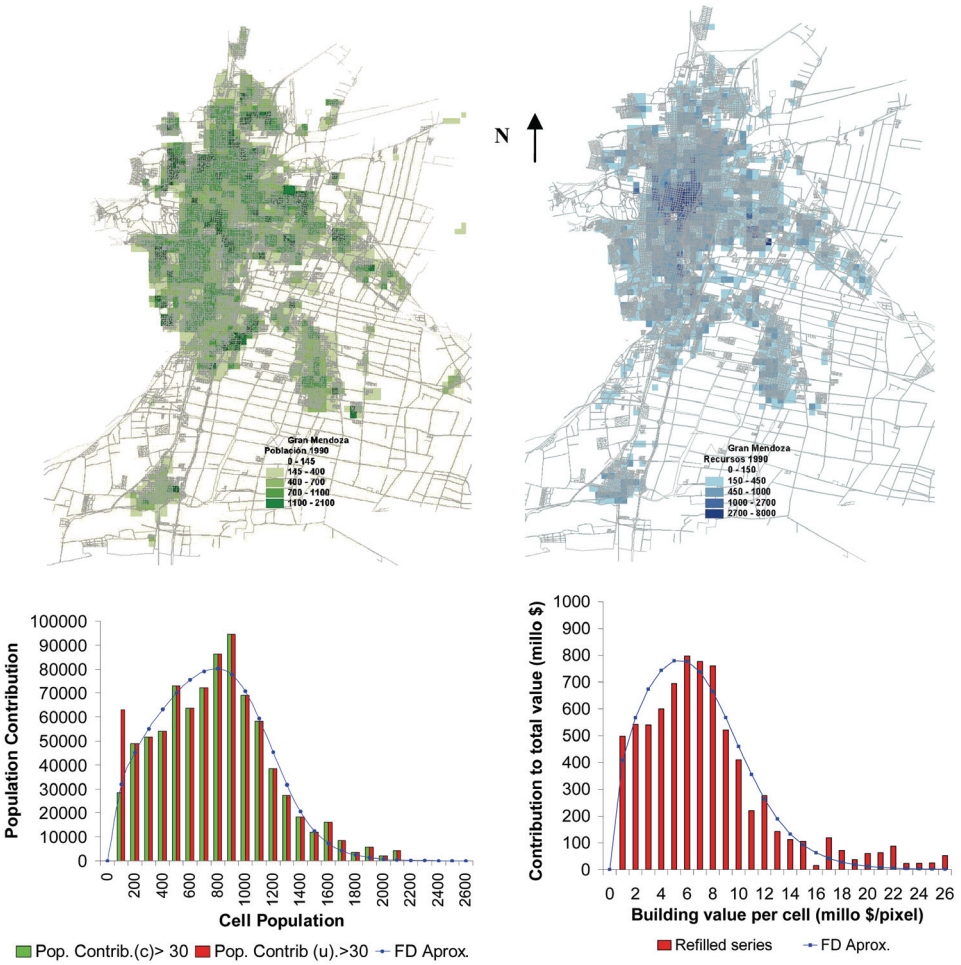


Fig. 3. Characterization of Great Mendoza– Above: GIS raster images of the population distribution (left) and of real estat values (right.)-. Below: Spectral densities of population and real estate and their approximations by Fermi-Dirac density distributions

$$F(N_p) = 1/\{1- \exp[(N_p-N_F)/N_T]\} \rightarrow \Delta N_P = A_p N_p^{1/2} F(N_p) \quad (14.a)$$

$$F(R_p) = 1/\{1- \exp[(R_p-R_F)/R_T]\} \rightarrow \Delta R_p = B_r R_p^{1/2} F(R_p) \quad (14.b)$$

where eq. (14.a) stand for inhabitants and (14.b) for recursons. The FD parameters are obtained within correlation factors of 0.98999 and 0.978 respectively (table 1).

Inhabitant (N_P)			Recurson (R_P)		
N_F: Fermi density [inhab/pixel]	N_T: Thermal density [inhab/pixel]	A_P: Density of states [inhab/pixel]	R_F: Fermi density [millon$/pixel]	R_T: "Thermal" density [millon$/pixel]	B_R: Density of states [millon$/pixel]
1140	165	3198	8.5	2.3	424.6

Table 1. Fermi-Dirac parameters for Great Mendoza 1990/2

The *crystallographic equivalent model* (a virtual substrate) is constructed over the properties of FD approximations (Kittel C., 1995):

$$\delta N_p/\Delta\varepsilon = 2/3\, N_p/\varepsilon \quad (15)$$

$$g_V(\varepsilon) = (1/2\pi^2)((2m_p)^{3/2} / \check{a}_p^3)\, \varepsilon^{1/2} \quad (16)$$

Equation (15) represents an associated energy state space (ε) with an $\Delta\varepsilon$ uncertainty, and eq. (16) the population volumetric density of states $g_V(\varepsilon)$ in this space (directly related to A_p in eq.14.a); here m_p represents an effective mass of the pseudo particle inhabitant, while $\check{a}_p = a_p/2\pi$ is an equivalent action constant proper of the urban process scale.

Fitting energy uncertainty to half of the excess of biokinetic energy over the daily rest metabolism of an inhabitant and the effective mass to the biokinetic proportion of its average mass, an effective mass of 12.83 [kg] and an action constant of 153.9 [J.seg] are obtained. Scaling can be completed assuming that in eq. (14.a) and eq. (14.b) a "thermal" equilibrium is fulfilled (stationary conditions) so that k. $N_T = R_T$, with k = 13939.39 [$/inhab] representing one recurson per inhabitant. In such case Br results in 30460 [rec/pixel], which permits to estimate the effective mass of the recurson as $m_r = 4.48657 m_p \cong 4.5\, m_p$

The band model can be finally obtained, taking into account that the recurson is the inhabitant's anti-particle, satisfying a representation analogous to a semimetal.

$$N_F / N_T = (E_F - E_{BP}) / E_T ; R_F / R_T = (E_{BR} - E_F) / E_T \quad (17)$$

Considereing the bottom of the population band is the zero energy level, the ceiling of the resources band is at 32 J and the Fermi energy level at 20,05 J .

3.2 Growth and circulation model

When representing the urban virtual substrate with a band structure analogous to a semimetal, it is possible to anticipate equivalent processes of conduction, as much of the free type as by movement of vacancies. The former can be associated mainly to fast dynamics on a daily basis, whereas the latter to the medium term transport that arises from expansion. Naturally, population dynamics adopts thence a similar transport model (circulation and growth):

$$\frac{\partial}{\partial t} p(x,y,t) = g_p(x,y,t,T) + 1/q_p \, \nabla . \boldsymbol{J}_p(x,y,t) \tag{18}$$
$$\frac{\partial}{\partial t} r(x,y,t) = g_r(x,y,t,T) - 1/q_p \, \nabla . \boldsymbol{J}_r(x,y,t)$$

where $[p(x,y,t);\ r(x,y,t)]$ represents the surface concentrations of both pseudo particles, inhabitants and recursons, $[g_p(x,y,t,T);\ g_r(x,y,t,T)]$ their speeds of growth and $[\boldsymbol{J}_p(x,y,t);\ \boldsymbol{J}_r(x,y,t)]$ the corresponding current densities, for which a *population equivalent charge* q_p will be defined afterwards. Besides, the growth part of the transport model specified in the pair of equations (18) follows the form:

$$\delta p = \eta_{0p} p_0 - \gamma \, p.r \, / \, p_0 \tag{19}$$
$$\delta r = \eta_{0r} r_0 - \beta \, \gamma \, p.r \, / \, r_0$$

where $\delta p = g_p.\Delta t$ and $\delta r = g_r.\Delta t$ represent the variations of concentrations of pseudo particles in a Δt interval, η_{0p} and η_{0r} their free growth rates, γ a factor of mutual control of population and $\beta = r_0/p_0$ an urban quality factor, with p_0 and r_0 the respective local stationary concentrations at statistical "temperature" T_0. Thus, the growth for each pseudo particle adopts the form of a balance between generation (production) and recombination (loss), as it could happen in the case of doped solid materials because of extrinsic excitation, a form that in addition can be linked to a prey-predator model typical of population dynamics in ecology (see for example: Bossel H., 1986; Pacala S. and Levin S., 1997). This requires the definition of population and resources growth rates, and of a recombination rate, that here is to be interpreted as a cross limitation to the free rates of growth.
As for the circulatory part of the transport model, this one follows the form:

$$1/q_p) \, \boldsymbol{J}_p(x,y,z,t) = -\mu_p \, . \, p . \nabla V + D_p \, . \nabla p \tag{20}$$
$$(1/q_p) \, \boldsymbol{J}_r(x,y,z,t) = -\mu_r \, . r . \nabla V - D_r \, . \nabla r$$

Currents adopt in each case the form of a dynamic balance between a drift current, mobilized by the gradient of an *urban potential*, and a diffusion current, mobilized by the gradient of the corresponding concentration. Currents demand the definition of an appropriate urban potential (in which the population charge mentioned above takes part), and a spatial tensor of mobility and diffusion $[\mu_p, D_p; \mu_r, D_r]$.
This type of transport and growth model is naturally attainable by means of bidimensional cellular automata of mobile agents, characterized by a set of parameters that are a function of space and of the statistical temperature of the system. Hereby, nevertheless, the additional advantage lays in that the analogy with the solid state of matter allows a more conceptual bottom-up interpretation, diminishing therefore the necessities of model parameterization to an indispensable minimum.

3.3 Urban potential and the population equivalent charge

From the point of view of the individual contribution of an inhabitant, the urban potential represents a measurement of its energy reserve, as a result of the capacity of the individual to collect resources from the environment, as seen already for the bioautomaton. Appart from what is stated in ec. (2), it can also be defined by means of a bottom-up approach analogous to the Thomas-Fermi model, used in solid state physics (Kittel C., 1995), taking

advantage of the scaling of characteristic constants already made in the band model; therefore, it is also possible to define the value of population charge.[4]
Assuming a monostructure of bands, a model of Thomas-Fermi adapted to the case can be specified as follows:

$$V_p(x,y) \cong (\check{a}_p^2/2m_p)\{(3\pi^2 p_{0m})^{2/3} - sgz \cdot (3\pi^2 \mid z(x,y) \mid)^{2/3}\} \quad (21)$$

$$z(x,y) = p(x,y) - r(x,y)m_p/m_r$$

The energy term given by $\varepsilon_{F0} = (\check{a}_p^2/2m_p)(3\pi^2 p_{0m})^{2/3}$ is the Fermi level for zero statistical temperature, where p_{0m} corresponds to the average concentration of pseudo particles inhabitants in their basal state (rest state), and $z(x,y)$ is the associated net concentration to the distribution of population charge ($r(x,y)$ is given in [rec.]) with its corresponding sign (sgz). From this theory and ec (2) , it is possible to find a suitable value for q_p . For the case study, a population equivalent charge $q_p = 5.832$ [(J.m)$^{1/2}$] can be found.

Fig. 4. Urban Potential for Great Mendoza (1990/92) in equipotential contour lines format.

Urban Potential in fig. 4, resulting from ec. (21), shows the centre of the city as a positive (dark) peak due to a bigger concentration of resources (Capital Department). Using this urban potential it is easy to distinguish metropolitan residential areas, resources injection areas, as well as the variation of urban quality and the relation between poles . The single

[4] For urban evolutionary situations being governed mainly by spatial nuclei of activity concentration, the Thomas - Fermi model allows the description of inhomogeneities in the distribution of population and resources, by means of smooth variations of the Fermi level, within a unique structure of bands (impoverished or enriched by resources and/or population). In evolutionary situations governed by fragmentation, the description must be made by zones with interphases that can present very steep transitions and even different band structures, altogether implying nonlinear local behaviours

reading of this potential map already gives substantial information of the city, thenceforth constituting a valuable synthetic way of representation in itself, even to the extent of a qualitative evaluation of future evolution.

3.4 Growth parameters

From the pair of equations (19) and since $p = p_0 + \delta p$ and $r = r_0 + \delta r$, for situations of normal growth in which $\delta p/p_0 << 1$ and $\delta r/r_0 << 1$, one gets (despising quadratic powers) the following relative variations of concentrations:

$$\delta p/p_0 \cong [\eta_{0p}(1-\gamma) - \beta\gamma\,(1+\eta_{0r})] \,/\, [1+\gamma\,(1-\eta_{0p})+\beta\gamma\,(1+\eta_{0r})] \qquad (22)$$

$$\delta r/r_0 = \eta_{0r} - (1/\beta)\,(\eta_{0p} - \delta p/p_0)$$

where the factor of urban quality $\beta = r_0/p_0$ is a function of space $\beta(x,y)$. Using this last property it is possible to fit the growth model to the case study, computing the average factor of urban quality by department β_{md} and associating them to the expected rates of annual population and GGP growth in the early 90's (Fig. 5)

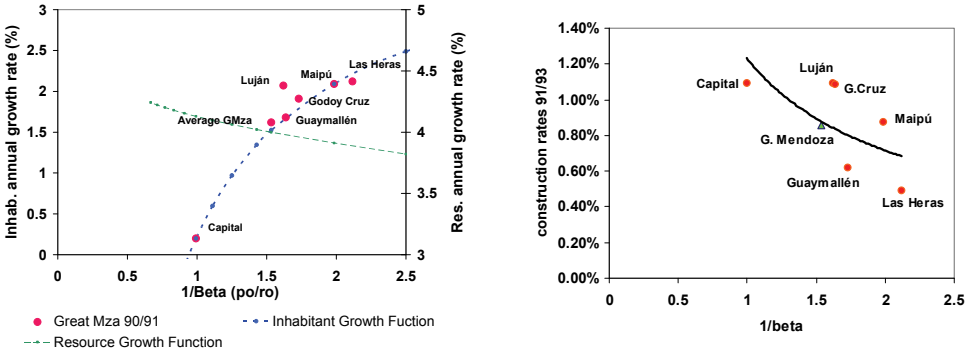


Fig. 5. Left: Demographic growth rates for Great Mendoza (90/91) as a function of the inverse quality factor $1/\beta$ and adjustment of the theoretical growth model . Right: Construction growth rates for Great Mendoza at the beginning of the 90s

The resulting growth parameters for the case study are η_{0p} = 4,2 % , η_{0r}= 8,1 % , and γ = 3,66%. The free demographic growth is an annual net birth rate η_{0p} (birth of inhabitants minus their mortality), representative of the effective procreative capacity. Similarly, the free economical growth rate η_{0r} represents a maximum average annual net growth of resources (here taken as real estate rates) since it results from the adjustment to the expected GGP growth.

It should be bared in mind that these growth parameters are representative of a stationary behaviour; other behaviours can arise out of the cultural substrate, which can modulate the balancing of population as much as through the parametric variation of η_{0p} as of γ . Likewise, the economical substrate influences on the balancing of resources through own parametric variations of η_{0r} and γ , but hereby depending more on macro economic conditions given at a national or a regional scale, rather than on a metropolitan scale. This justifies the need to represent them as functions of statistical temperature.

3.5 Mobility and diffusion factors

The circulation part of the transport model is written in the pair of equations (20). For deriving the mobility and diffusion factors one can consider that the system has reached a stationary situation where $J_p(t=0) \cong 0$ and $J_r(t=0) \cong 0$, then:

$$D_p / \mu_p \cong p_0 \left| dV/dp_0 \right| ; \quad D_r / \mu_r \cong r_0 \left| dV/dr_0 \right| \tag{23}$$

As the urban potential V is approximated by Thomas-Fermi, one gets a generalized expression of D_p/μ_p as a function of space [5]:

$$D_p/\mu_p \cong \frac{2\,\varepsilon_P(x,y)\left|1-(dr/dp)_0\, m_p/m_r\right|}{3\,q_p\left|1-(r_0/p_0)\, m_p/m_r\right|^{1/3}} \tag{24}$$

where $\varepsilon_P(x,y) = (\check{a}_p^2/2m_p)(3\pi^2 p_0(x,y))^{2/3}$ can be considered the isolated contribution of $p_0(x,y)$ to Fermi's level. The former applies for genuine stationary conditions, but for a quasi stationary frame there should be a limiting trend as follows:

$$D_p/\mu_p = D_r/\mu_r \cong 2/3\ \varepsilon_F(x,y)/q_p \tag{25}$$

Once the D/μ quotient has been specified for each pixel, a numerical value of each parameter can be found by estimating mobility factors, as in solid state theory:

$$\mu_p = q_p\,\tau_p / m_p ; \quad \mu_n = q_p\,\tau_r / m_r \tag{26}$$

The characteristic time parameters τ_p and τ_r, can be interpreted as the *average free time periods between relocations* of inhabitant and recursons. In the case of τ_p, its value is representative of the average time invested daily per inhabitant in terms of displacements (in one direction) for different activities, which for the case study was about 25 min in 1990 [6].

An average measure of D/μ quotient is the given by Einstein's equation $D_{p0}/\mu_{p0} = D_{r0}/\mu_{r0} = KT/q_p\ [(J/m)^{1/2}]$, being for the case study $\mu_{p0} = 1{,}15\ 10^{-04}$, $\mu_{r0} = 2{,}59\ 10^{-05}$ for the effective mobility factors $[(J.m)^{1/2}\ sec/kg]$ and $D_{p0} = 5{,}91\ 10^{-5}$, $D_{r0} = 1{,}51\ 10^{-5}$ for the effective diffusion factors [m2 /seg]. Factor D_{r0} in particular, can be interpreted as the city average "thermal" expansion, gives a relative surface expansion of 1.65. Since this is practically $(1+\beta_0)$, where $\beta_0 \cong 0.65$ is the city average quality factor, it gives a net relative expansion of β_0 per inhabitant in excess, which is in accordance to the fact that excess concentration will be rearranged trying to conserve the former average quality factor.

The mobility and diffusion factors link the daily commutation regime to the expansion regime, in a way that the city structure depends directly on the average relocation time period and vice versa.

3.6 Implementation and testing of the model

The general scheme of calculation associated to the model, consists of an iterative process of n periods (annual periods have been used for the case study), in which, as from an initial

---

[5] Mobility and diffusion factors vary in space and hence do not follow the analogous relationship to Einstein's equation $D/\mu = kT/q$. For an explanation on Einstein's equation see for example Kittel C., 1995.

[6] Weighted sum of invested time in bus and car journeys not including Luján de Cuyo.

state, the urban potential and the growth and transport of population and resources are computed for each cell (64x 86 elements of 350x 350 m2), thus filling in an evolutionary gridded data base. The parametric inputs of the model are given by the growth free rates and mutual control of population and resources, as well as by their respective mobilities.

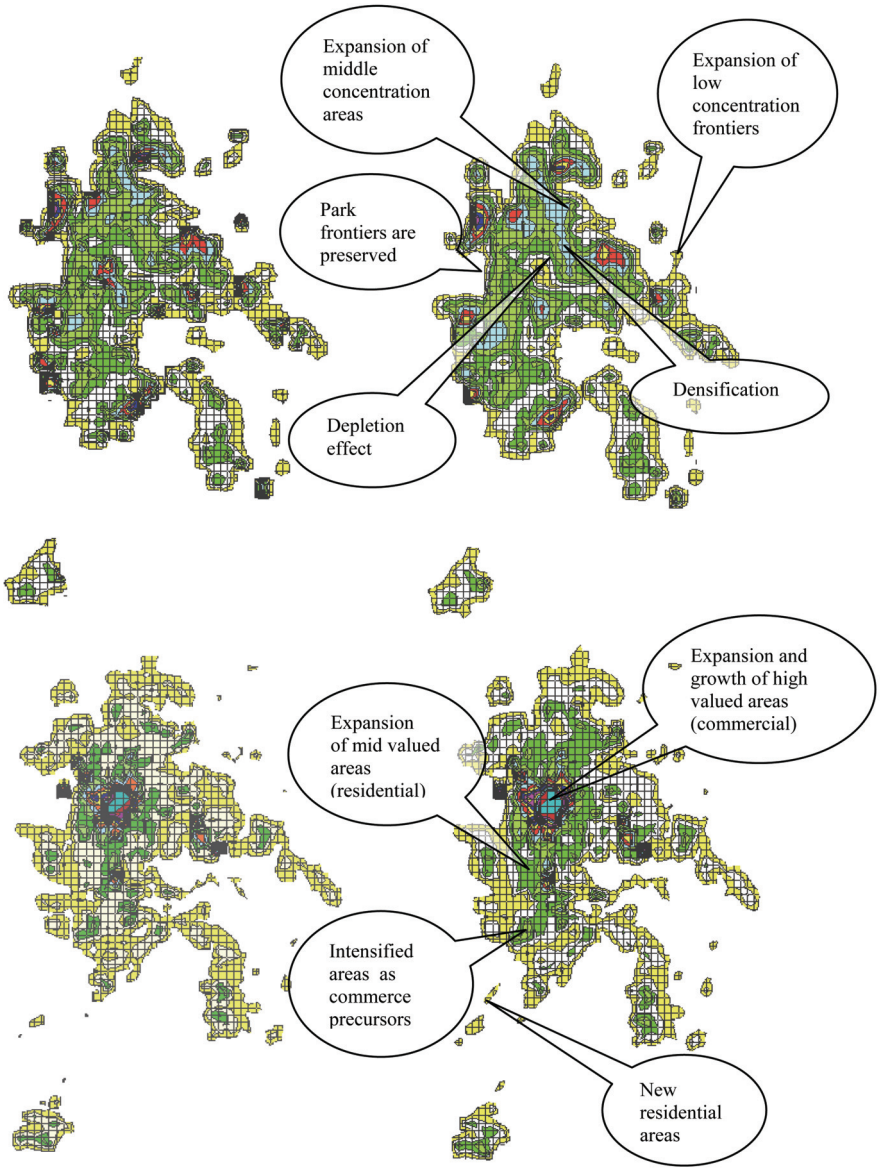


Fig. 6. Comparison between 1990/1 (left figure) and final state (right figure) for the distribution of inhabitants (top figures) and of recursons (bottom figures).

Spatiotemporal uncertainty, associated to the initial state, limits the model space and time resolution and might even cause its instability. It includes the combined effect of errors due to the gathering, sampling and conditioning of demographic and cadastral data, which on the other hand are not strictly co-temporal. For the case study the time uncertainty was lower than 1.4 year and a space uncertainty not larger than one pixel (350 x 350 m^2).
This model has been tested for the case study in quasi stationary conditions. Initial data correspond to Great Mendoza in 1990/1, with the associated characteristic constants previously discussed. Parameters have been kept constant throughout all periods.
The results of a simulation for five years show a good correspondence with growth and distribution trends seen in such decade. From the maps one can clearly distinguish how equidensity areas evolve (fig. 6). Only considering the spatial aspects observed here, one can already recognize the principal types of effects that could be expected in mid-term evolutions in any city, as for instance the one conveyed by the seven transition rules of the Batty-Torrens model (Batty M., Torrens P., 2001). It is particularly interesting the depletion effect (sometimes called "donut" effect) seen in the main centre of the city, which here arises naturally as a consequence of resources and inhabitants competitive growth and diffusion. This qualitative correspondence to the principal trends of evolution of Great Mendoza during the last decade, acquires more importance when considering it together with the reasonable overall temporal behaviour of state variables.

4. Some reflections on population growth and economy

Since the beginning of the last century the world is experiencing an important demographic transition, which will probably impact on economic growth. Many demographers and social scientists are trying to understand the key drivers of such transition as well as its profound implications. A correct understanding can help to predict other important trends at global scale, as the primary energy demand and the carbon emission to the atmosphere, which may be leading to an important climate change.
Inspired on the former works, a set of coupled differential equations has been proposed in Puliafito S. Enrique et al. (2007) to describe the changes of population and gross domestic product, modelled as competing-species as in Lokta-Volterra relations. In fact, if the development and population dynamics of cities could be explained in terms of the above given model, it would be natural then to expect that global population growth and economy follow also a predator-prey type model (eq. 19). Based on that, changes of primary energy consumption and carbon emissions would be then modelled similarly. The estimated results for the temporal evolution of world population, gross domestic product, primary energy consumption and carbon emissions were calculated from year 1850 to year 2150. The calculated scenarios are in good agreement with common world data and projections for the next 100 years.
Economic growth models give population growth a major role, but some show population as detrimental to economic growth and others show population as a major contributor. In fact, population growth has two effects: it increases the number of consumers, and it increases the number of workers devoted to productive activity and research. However, population growth increases the scale of the economy, therefore permitting industries, enterprises, and the entire economy to exploit economies of scale. Models based on technological progress, or on generation of new ideas generally conclude that population growth and the size of the population have a positive effect on growth of per capita output

by specifying technological progress as a function of the number of people engaged in R&D activity. But models based on congestion, come to the conclusion that increasing population produces a slowing economy, since more investment is needed to maintain same per capita output. The debate on whether population growth is detrimental or beneficial to the welfare of humanity essentially comes down to the opposing conclusions of the Solow and Malthusian models vs. the exogenous growth models (Galor, O., Weil, D. , 2000).
The definition of economic growth as an increase in output per capita implies an inverse relationship between output (GDP) and population, but this is not necessarily a cause-effect relationship; if population causes total output to increase faster than population does, only then it will produce an increase in per capita output. Although in many countries population growth seems to be negatively related to economical growth, empirical evidence does not unambiguously support either view of population growth.
For a closer look on this, consider population p when changes are taken as continuous and are unregulated by external factors; then it can be expressed in differential form as:

$$1/p\ (dp/dt) = \eta \tag{27}$$

which gives as solution a growing exponential function of the type $p(t) = P_0 exp\ (\eta t)$, where η is the growth rate. However, many demographic and ecological studies recognize that, for long periods of time, the growth rate η is not constant, but decreases as population increases. So the actual population presents apparently a (auto-) limiting factor. In fact, this limitation can be expressed as in differential form as:

$$1/p\ (dp/dt) = \eta - \alpha\ p \tag{28}$$

where the crude growth rate η is limited by the product of $\alpha.P_t$, being $\alpha = \eta/P_m$, and P_m the maximum supporting population for a given environment, which produces the "S-shaped" curve, known as logistic curve. Also the economic output (GDP) sometimes is modeled in a logistic form. Although population and gross domestic product may be fitted to logistic type curves, there is no clear indication on which may be the value of the maximum carrying capacity, nor a clear explanation for this limitation process. One possible feedback mechanism, which may explain this limitation processes is linked to the availability of resources, as it can be seen from ecological and biological studies and the discussion given in the former points. Consequently, a pair of nonlinear-coupled differential equation, similar to the Lokta-Volterra relations for two species interaction, is proposed:

$$\begin{cases} 1/p\ (dp/dt) = a - g\ m \\ 1/g\ (dg/dt) = \kappa - b\ p \end{cases} \tag{29}$$

where the left members represent the relative changes in the population p and available resources g, $b.p$ is the annual resource consumption by the population p, k is the annual resource renovation, m is the annual death rate, a is the per capita consumption and regulates the birth rate n. Interesting to note is that depending on the chosen parameters, these coupled no linear relationships may show a chaotic behavior. Eq (29) shows that for low values of g population will increase rapidly regulated only by mortality rate m, but as p growths the GDP growths is slowed down by increasing p, which in turn will slow down the population growth.

If p and g have similar temporal variation, which corresponds to a stationary frame where the ratio g/p (per capita output) is approximately constant, it is possible to foresee that p and g will also produce a logistic type equation. However, as for non stationary frames, the ratio of g/p is not constant, the logistic type curve can only be achieved if also a and b are not constant but they have the proper variations. To represent these types of frames adequately (particularly the transitory in short terms), an additional function $f(t)$ can be included to the set of Eq. (29), which might be interpreted as an external excitation function comprising all other causes of variation not included in the predator-prey solely mechanism; in fact, the Lokta-Volterra model is a closed one because the eventual changes in the carrying capacity of the substrate are not explicit. To make them explicit, considering now an open model, the substrate has to be taken as varying along the time, for example due to the changing culture and technology. To generalize this open model, disregarding if it is expressed in terms of the rates of production or consumption of the species, and at same time to capture the influence of the variation of the substrate as rates over the populations of the considered species, we can write:

$$\begin{cases} 1/p\ (dp/dt) = \alpha_1.f/p + \alpha_2.g + \alpha_3 \\ 1/g\ (dg/dt) = \beta_1.f/g + \beta_2.p + \beta_3 \end{cases} \tag{30}$$

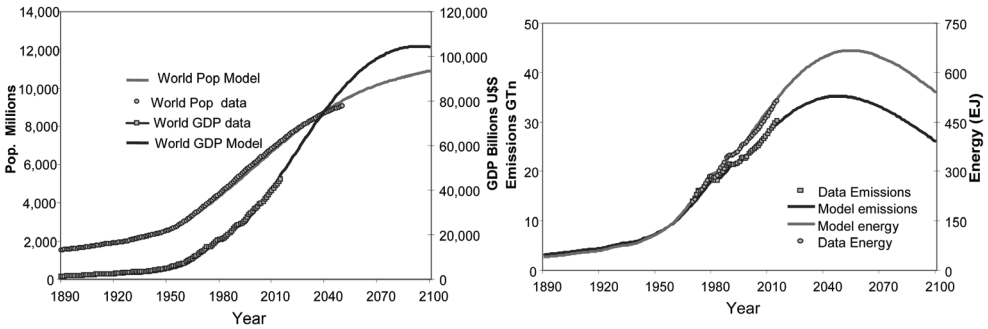


Fig. 7. (Left) Exogenous model for world population (millions inhabitants) and GDP (Billions U$S); (Right) Exogenous model for world primary energy consumption (EJ) and carbon emissions (GTn) (from 1890 to 2004 measured or estimated values; from 2005 to 2100 projected values). Coefficient values used in Eq. (30) are p_1=0.0004, p_2= -7.4×10^{-8}, p_3= 0.64%, P_0=1522; g_1= 0.0014, g_2= -2.5×10^{-6}, g_3= 1.68%, G_0=1234); coefficient values used in Eq. (31): ε_1=0.0009, ε_2= -2.8×10^{-6}, ε_3= 1.64%, E_0=40; σ_1= 0.0009, σ_2= -2.8×10^{-6}, σ_3= 1.45%, C_0=3). The external function $f=A.exp(\tau.t)$ plus short impulses is used to represent big international crisis with A=2, τ=0.04 from 1890 to 1963, and τ=-0.04 thereafter. Sources of data EIA (2005).

The experience shows that most positive culture and technology changes arise from scenarios with an increasing g/p rate; therefore, a first approximation is to set f equal to g/p. The figure 7 shows satisfactory results for Eq (30) in such condition; the coefficients α_1, α_2, α_3 and β_1, β_2, β_3 are obtained from the annual changes applying a multi-linear regression.
The annual changes in energy consumption and carbon emission show similar behaviour as changes in GDP and population. Despite that there is not enough certain information of carbon emissions and energy consumption from 1890 to 1970, the energy demand e and

carbon emission c are strongly coupled to g and p, so that a similar set of differential equations as (30) can be suitable to estimate the annual changes in both variables:

$$\begin{cases} 1/e\ (de/dt) = \varepsilon_1.f_1/e + \varepsilon_2.p + \varepsilon_3 \\ 1/c\ (dc/dt) = \sigma_1.f_2/c + \sigma_2.p + \sigma_3 \end{cases} \quad (31)$$

where $f_1=f.e/g$; and $f_2=f.\ c/g$, and f is the same function used for the external excitation of g and p in Eq. (30), for the exogenous model; $\varepsilon_1.\ e/g$ (%) is the efficiency improvement through more technological investment, $\varepsilon_2.\ p$ (%), is the per capita energy consumption, and ε_3 is the residual increase in energy consumption not explained by the other two coefficients, or the natural increase without an external excitation. Same can be said for the natural rate of changes in the carbon emissions. Some results are also shown in Fig. 7.

5. Conclusions

Throughout this chapter we have been exploring some of the fundaments of CA models and the reasons of why these are being so widely applied nowadays, particularly to urban systems and ecology, all of which seems to be connected directly with the fact that the transport equations are common as much to the socioeconomic phenomena as to physics. However, it is not immediate that population dynamics can be described similarly by means of reaction-diffusion equations; on the contrary, perhaps on this outstanding fact rests one of the clues to explain how individual behaviour, usually seen at the "microscopic" scale as mostly stochastic or eventually moved by free-will, can fit into the associated collective behaviour seen at the "macroscopic" scale.

In this sense, by means of the bioautomaton theory we have seen that the discrete character of the device-environment interaction, leads to describe stationary individual behaviour in a similar way to what is done in quantum stochastic systems. The most important aspect of this similarity is that the statistical behaviour of a bioautomaton can be represented in average by means of wave functions, in a way that stationary or quasi-stationary solutions regarding group behaviour can result from the superposition of individual wave functions. As a wave function is a measure of the probability of a stationary exchange between each device and its immediate surrounding, periodical spatial structures can emerge in certain conditions; hence, stationary dynamics would be described in terms of transport equations framed within some appropriate band theory.

This theoretical speculation is justified for real ecological systems when we consider that social behaviour and complex and regular spatiotemporal structures emerge under conditions where species reach some critical spatial density, thus giving place to outstanding interaction mechanisms as templates, stigmergy and self-organization. This suggest that an ecosystem is not the mere association of interactions as a whole or a collection of highly interactive independent elements, but a rather coherent sum of elementary units composed of living individuals and their near-by space-environment, the latter being regarded as a multidimensional representation of the necessary resources for survival, physical space in itself included.

Within this context, a high-density ecosystem can be compared in some sense to an elastoplastic network and thence treated in a similar way to the solid state of matter; that is, as state transitions in a pseudo-crystalline virtual substrate subdued to a general exclusion

principle. In fact, at describing the spatiotemporal dynamic evolution of populations of real individuals through transport equations, one is not only considering the interactions of the species with its space/environment, but also stability regions in the associated state space that are similar to the energy bands in solid materials.

Standing on these principles we have reviewed a feasible model for urban evolution, which is outlined in terms of a virtual substrate with two energy bands: a population band and a resource band where their associated pseudo-particles – the inhabitant and the recurson – represent (in principle in an anti-symmetrical way) the interacting population and their space- environment structure correspondingly.

The characterization of the energy band model for Great Mendoza, starts of the statistical properties of spatial distribution of inhabitants and of real estate values, which have been assimilated to Fermi-Dirac statistics; after determining the characteristic parameters of associated pseudo particles and of the band structure in itself, the case study can be represented in an analogous form to a semimetal.

Taking advantage of the solid state picture, the net concentration of pseudo particles can be linked to a proper urban potential function, through the use of a Thomas –Fermi approximation and an equivalent population charge. Thereafter, a static combined representation of the urban region is feasible in terms of the field theory.

With these elements, the dynamics of urban systems can be constructed over cellular automata with mobile agents, by using similar transport equations as in solid state. The circulatory part of the model adopts the balance form between two components (diffusion and drift), describing the concentration and sprawl of population and resources present in the cities. The model production part , described in terms of generation-recombination of pseudo particles, is comparable to a predator-prey model as well, typical of population dynamics in Ecology. Using the characterization of pseudo particles it is possible to adjust the diffusion and mobility coefficients, and growth to the well-known urban behaviour, with a "bottom-up" approach that diminishes the need of parameterization to an indispensable minimum.

A test in stationary conditions along a five-year period, shows that the principal state variables of the case study evolve in time as it would be expected from the application of classical methods based on statistical progression, and with a spatial response compatible with the principal effects expected in a mid-term evolution in a city. In this sense, this analogy plausibly explains the varied growth rates of the political departments, as well as the principal urban trends for spatial occupation for Great Mendoza in the last decade.

The methodology and model here discussed open new possible ways of approaching urban evolution. Although it has been presented as a stand-alone tool, it can be combined through its parametric inputs with other CA models (i.e. in successive embedded scales or lower structural bands) or even with non spatial social-economical models, thus orienting it more to long-term simulation, where innovation and changing scenarios are required. It also provides a way for describing fragmented urban development by means of zones which may have very different band structures, implying non-linear local behaviours in the resulting interphases.

Finally, a global perspective of the former ideas has been presented in the context of a research of the projection of the energy demand, the carbon emissions and the link to possible climate changes. Several authors have proposed that world population, the primary energy demand and the gross domestic product are the main drivers (or state variables) for the carbon emission problem, while per capita consumption, energy intensity and emission efficiency, among others, are taken as indicators of the system.

As the development and population dynamics of urban regions is represented by transport equations that include a production part , described in terms of generation-recombination of pseudo particles representing population and resources, it seems natural to expect that global population growth and economy follow also a predator-prey type model. Based on that, changes of primary energy consumption and carbon emissions can be then modelled.
Here we have seen that a set of coupled differential equations of this type can describe the changes in the main state variables in a plausible way. Indeed, some studies have observed both positive and inverse relation between population growth and GDP, depending on the time frame and the group of countries involved in the studies; with the coupled model here shown is possible to represent well the three different scenarios or transitional phases from "Malthusian, post Malthusian and modern growth", proposed by some scholars. Other researches propose logistic variation of the population as a way to describe the demographic transitions. Here, the interrelation between these variables, the growth rate and their expected logistic type shape curve arises naturally as the interaction of population and economic output as described in the coupled differential equations. The results of the model were compared to several agencies projection, showing comparable results, but most importantly is the ability to capture conceptually and mathematically the range of current thoughts and models used by the international agencies.
Cellular Automata have shown a great potential for modelling a wide range of types and scales of phenomena, but it is still an open question why this is so. A research on the foundation of this capability, as the one intended here, might contribute not only to a better understanding of the principles involved but also to a better and wider use of the tool.

6. References

Angulo J.M, Ruiz Medina M.D, Anh V., 2001: "Space-Time Fractional Stochastic Diffusion", in Mateu J. & Montes F. (ed.) *Spatio-Temporal Modelling of Environmental Process*- Castelló de la Plana: Universitat Jaume I.- Spain.

Ball P., 1998: "*The self-made tapestry*"- Oxford University Press

Batty M., Torrens P., 2001: "Modeling Complexity: The Limits to Prediction" *CYBERGEO*, No 201, 04 décembre 2001

Batty, M, 1996; Visualizing urban dynamics- in Longley, Batty (eds) *Spatial Analysis: Modelling in a GIS Environmen*t, 297-320; John Wiley and Sons- United States

Bossel H., 1986- *Ecological System Analysis: An Introduction to Modeling and Simulation*- German Foundation for International Development (DSE) and Food and Agriculture Development Center (ZEL)- Fed. Rep. Germany

EIA, 2005: *International Energy Outlook 2005*, Energy Information Administration, www.eia.doe.gov/oiaf/ieo/index.html

Galor, O., Weil, D., 2000- Population, Technology, and Growth: From Malthusian Stagnation to the Demographic Transition and Beyond- *American Economic Review*, September 2000, 90 (4), pp. 806-828.

Gunter, B., Gonzalez, U., Morgado, E., 1992- Biological Similarity theories: a comparison with empirical allometric equation. *Biol. Res* 25 (7-13)

Hemmingsen A.M, 1960- Energy Metabolism as related to body size and respiratory surfaces, and its evolution- *in Rep.Steno Mem. Hosp*. 9, 1960, pp 1-110

J.F Nystrom , 2001; Tensional computation: Further musings on the computational cosmography- *Applied Math. and Computation*-Vol. 120, 1-3, pp. 211-225 – Elsevier

Jeanson R, Blanco S, Fournier R, Deneubourg JL, Fourcassie V, Theraulaz G., 2003- "A model of animal movements in a bounded space"- *Journal of Theoretical Biology* 225, p. 443-451-Elsevier

Kittel, C., 1995- *Introducción a la Física del Estado Sólido* – Ed. Reverté España (Introduction to Solid State Physic, Sixth edition by John Wiley & Sons Inc.)

Klieber M., 1961 - *The Fire of Life*- John Wiley & Sons Inc., New York 1961

Lebiedz D. and Brandt-Pollmann U., 2003: "Manipulation of Self-Aggregation Patterns and Waves in a Reaction-Diffusion System by Optimal Boundary Control Strategies"- *Phys. Rev. Lett.* 91, 208301

Meinhardt, H., 1982 : "Models of biological pattern formation"- *Academic Press*, London 1982

Mitas L., Brown W. M., Mitasova H., 1997: "Role of dynamic cartography in simulations of landscape processes based on multi-variate fields", *Computers and Geosciences*, Vol. 23, No. 4, pp. 437-446- Elsevier

Mitasova, H; Mitas L. 2000: "Modeling spatial processes in multiscale framework: exploring duality between particles and fields". *Pl. talk at GIScience2000 conference*, Savannah.

Nelson, E., 1966, -Derivation of the Schrödinger Equation from Newtonian Mechanics- *Physical Review* 150, 1079–1085

Pacala, S.; Levin, S.. 1997. Biologically generated spatial pattern and the coexistence of competing species. Tilman, D; Kareiva, P. (eds.) Spatial Ecology: *The Role of Space in Population Dynamics and Interspecific Interactions.* Princeton Univ. Press; pp 204-232.

Park S. and Wagner D. F. , 1997: "Incorporating cellular automata simulators as analytical engines in GIS, *Transactions in GIS,* 2(3), 213-231- Wiley-Blackwell

Popov V.L., Psakhie S.G., 2001 "Theoretical principles of modeling elastoplastic media by movable cellular automata method. (I)" - *Physical Mesomechanics*, 4 I 15-25-Elsevier

Puliafito, José .L.; Puliafito S. Enrique 2007- Bioautómatas: Dispositivos Autónomos de Comportamiento Estocástico con Similitud Cuántica - *Mecánica Computacional Vol. XXVI* pp.3418-3439- S. Elaskar, E. Pilotta, G. Torres (Eds.); Córdoba, Argentina,

Puliafito, José Luis 2006- A transport model for the evolution of urban systems-*Applied Mathematical Modelling* 31 (2007) 2391–2411- Elsevier

Puliafito, S. Enrique, Puliafito José Luis, Conte Grand, Mariana, 2007- Coupling population dynamics to carbon emissions-*Ecological Economics* -Volume 65, Issue 3, April 2008, Pages 602-615 - Elsevier

Smolin, Lee 2007- Could quantum mechanics be an approximation to another theory?- *arXiv:quant-ph/0609109v1*

Theraulaz G., Gautrais J., Camazine S.,. Deneubourg J.L, 2003; "The formation of spatial patterns in social insects: from simple behaviours to complex structures"; *Phil. Trans. Royal Society*, A 361, p. 1263-82, 2003

Torrens P., 2002: "How cellular models of urban systems work. (1.theory)"- *CASA Paper* 18 – Centre for Advanced Spatial Analysis- University College London

Wikle C., 2001- "A kernel-based approach for Spatiotemporal Dynamic Models", in Mateu J. & Montes F. (ed) *Spatio-Temporal Modelling of Environmental Processes*- Castelló de la Plana: Universitat Jaume I.- Spain

Part 2

Dynamics of Traffic and Network Systems

8

Equilibrium Properties of the Cellular Automata Models for Traffic Flow in a Single Lane

Alejandro Salcido
Instituto de Investigaciones Eléctricas, División de Energías Alternas
Mexico

1. Introduction

In the last thirty years, the application of cellular automata as models of physical systems has attracted much attention, particularly for studying and simulating behaviour of fluid systems and traffic flow. In this work we present a theoretical analysis of the equilibrium properties of the cellular automata models for multi-speed traffic flow in a single lane highway. We hope our studies may advance some steps in the line of establishing a quite well formulated physical theory for these models. Our interest in this problem comes from the believe that general theoretical results about the traffic cellular automata may help very much to improve the speed of the associated computer models that scientists and engineers use for traffic flow simulations; but on another hand, it is much close related to the need of having, in a near future, a simple, but efficient tool for estimating the distribution in space and time of the pollutant emission rates coming from vehicular traffic in urban settlements, in such a way we can use the simulation results as the emissions input for the air pollution dispersion models we use to asses air quality in big urban places like Mexico city.

1.1 Motivation background and antecedents

The general development of human societies settled down in urban sites has given new dimensions of all kinds to air pollution the world over during the last few decades. For cities that have became (or are becoming) into geopolitical centres of urban regions with high economic activity, air pollution is a fast growing problem because of the increasing urban population causing high densities of motor vehicle traffic, greater electric power generation needs, and expanding commercial and industrial activities. The high volumes of emissions released to the atmosphere from the urban settlements have such a significant magnitude that a healthy air quality cannot be achieved by natural regeneration (or homeostasis) and scavenging processes only.

A major problem causing high levels of air pollution in big urban settlements, which also increases the complexity of analysis and evaluation of air quality, is the fossil- fuelled urban transport system and its interaction with the city, because motor vehicles produce different emissions under different driving conditions of speed, acceleration and idle. Traffic problems are, in fact, the main culprits of air pollution in urban areas, but that is not the end of the story, because their impacts actually extend even further. The intense traffic of motor vehicles, and their recurrent congestion and jamming produce waste of time and money, increase the risk of car crashing, promote the social unrest, and produce high stress levels

and health deterioration of the inhabitants of the cities. On another hand, urban traffic is a very complex problem. A growing number of the metropolitan areas world-wide are suffering a transportation demand which largely exceeds capacity. But in many cases, unfortunately, a good enough solution or, even desirable is not simply to extend capacity to meet the demand. Nowadays a coherent handling of the large, and distributed, transportation systems has become in a priority issue in urban planning and management.

Pollution from traffic consists of particulate matter, nitrogen oxides (NO_x), carbon monoxide (CO), volatile organic compounds (VOCs), sulphur dioxide (SO_2), and also other compounds in small amounts, like polyaromatic hydrocarbons (PAH). Lead emission from traffic has reduced dramatically after moving to unleaded fuels. Particulates come from the exhaust emissions, especially from diesel engines, but also from dust and dirt from roads and tires. Other fine particulates are formed by chemical reactions in the atmosphere. Formation of ozone O_3 in urban areas is mainly caused by traffic pollutants in a photochemical reaction with UV radiation from the sun.

In the Mexico City Metropolitan Area (MCMA), which is composed of 16 delegations of Mexico City and 59 municipalities of the State of Mexico, the registered vehicular fleet is estimated at more than 4.2 million vehicles. Among these, the 62% are vehicles registered at Mexico City and the remaining 38% are units which belong to the State of Mexico. In these figures, private cars account for a significant percentage (80% in 2006) of the units for the transport of people, and they constitute the most polluting category, producing 52% of the CO, 33% of the NO_x, and 21% of the SO_2 that are released to the MCMA´s atmosphere. Diesel vehicles, particularly trucks and buses, are other important emission sources which contribute, respectively, with the 28% and 16% of $PM_{2.5}$ (particles with diameters < 2.5 μm), and altogether, with 21% of NO_x (SMA-GDF, 2008).

Against this background, it becomes quite relevant the prediction of the air pollution caused by vehicular sources in urban areas. For this purpose, it is quite important to be able to estimate the space-time distribution of the motor vehicles moving inside an urban area, because, as a matter of methodological order, it is a prior step in estimating the space-time distribution of its respective air polluting emissions. Moreover, since changes in the urban morphology and the spatial distribution of the build in a city can affect traffic flow, and thus the space-time distribution of the vehicle emissions, for the purposes of studying the urban air pollution problems, as well as for the city development planning, it is a quite relevant issue the searching and the developing of simple and reliable tools for simulating vehicular traffic, its emissions and their impacts on air quality. At the MCMA, this is important because, in addition of the daily intense traffic, the urban morphology has changed significantly in last decade, specially by the construction of second floors on the main traffic corridors and explosive growth in the number of skyscrapers and other big buildings.

Air quality analysis for mobile sources, in most cases, is a very complex process which is performed by a combination of several different models. However, although sometimes these models are considered independent of each other, such as it occurs when simply we take the output of one of the models as the input of next one in line, in the real world we find dynamical couplings between processes or phenomena, which cannot be ignored completely in their modelling. Then, it is important to take into account the possible dynamical couplings between the models in integrating the enveloping computational package (or final model) of analysis.

There are basically three types of models required to perform the analysis of the impact of traffic on air quality, as it has been illustrated schematically in Fig. 1. The first type is the

model describing and projecting the vehicle activities of the facilities to be analyzed. In general, professionals of transportation use two modelling scales, transportation planning models (interested in regional analysis) or traffic flow models (interested in local transportation facilities such as individual roadways, intersections, and ramps, etc.). The second type of analysis (emissions rate models) represents the process of estimating emissions by vehicle fleets. When emission rates are combined with vehicle activity data, the result is an estimate of emissions by time and space. Once the vehicle activities are estimated, and combined with the emissions rates, the atmospheric dispersion of the vehicle emissions can be estimated with a pollution dispersion model. This third type of analysis is performed to estimate pollution concentrations. This final modelling step is needed to estimate pollutant concentrations to which humans are exposed. In this analysis, temporal and spatial estimates of pollutants from transportation and other sources, along with estimates of background pollution and meteorology conditions, are combined. When this analysis is completed, comparisons of estimated pollution concentrations with the National Ambient Air Quality Standards are made to determine whether control action is needed.

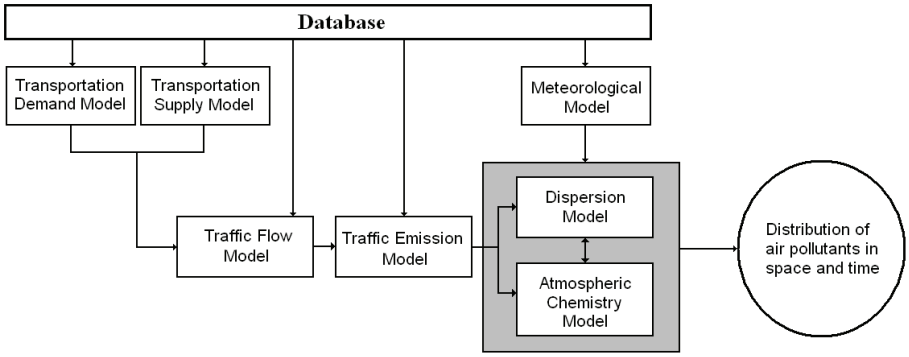


Fig. 1. Simplified combination of models for estimating the impact of traffic flow on air quality. In real world there are feedbacks that complicate the system, such as the influence of transportation supply on land use, or the effect of traffic flows on travel demand in case of congestion.

The knowledge of the wind circulation events constitute an important issue for estimating and understanding how the emissions of air pollutants will be dispersed in an urban settlement and how the air pollution of a city may be exported towards neighbouring sites. Some local pollution dispersion models require only some limited meteorological inputs, but some others may require a detailed knowledge of the wind field, for example. The theoretical basis of meteorological models is in the Navier–Stokes equations, which constitute a system of coupled and non-linear partial differential equations (Batchelor, 1967). For small velocities, these equations can be linearised and solved without much difficulty, analytically if the solid boundaries involved are simple, and numerically otherwise. However, when air flow velocities are large, instabilities may appear and exact analytical methods can no longer be used. Even numerical methods are difficult to use, chiefly because scales of different sizes must be taken into account, which forces grids either to be very small or variable. In practice, a lot of powerful computer simulation tools for diagnostic and prognostic purposes can be found for a variety of applications where the wind field and

other meteorological variables are required. Notwithstanding, the numerical solutions depend strongly on boundary conditions and initial values; so that special care must be taken to correctly initialise all meteorological variables in the computational domain and to correctly define the time-varying physics at the boundaries (Zannetti, 1990). Two excellent prognostic meteorological models are the PSU/NCAR mesoscale model (MM5) and the Weather Research and Forecasting (WRF) model. These models are complex and heavy numerical simulation instruments adequate only for mesoscale problems (MM5, 2003).

In what concerns to the traffic flow modelling, on the other hand, two different conceptual frameworks are used in general. The oldest one is based on a coarse-grained fluid dynamical description, where traffic is modelled as the flow of a continuum vehicle gas (Kühne, 1998). The other framework is that one of the microscopic traffic models. Here, attention is explicitly focused on individual vehicles, each of which is modelled as a particle that may interact with each other, affecting the others movement. Within this framework, the dynamical evolution of the vehicle gas has been described in terms of several different types of mathematical formulations. For example, a probabilistic description of vehicular traffic has been proposed based on appropriate modifications of the kinetic theory of gases (Prigogine & Herman, 1971; Helbing, 1996, 1998; Helbing & Treiber, 1998; Nagatani, 1997a, 1997b), while a deterministic description is provided by the car-following theories based on the Newtonian mechanics (Herman & Gardels, 1963; Gazis, 1967; Rothery, 1998).

Like the molecular approaches of computational fluid dynamics, microscopic simulation of traffic flow phenomenon has always been regarded as a time consuming complex process involving detailed models that describe the behaviour of individual particles (such as molecules, in the first case, and motor vehicles in the second one). Nevertheless, in the last three decades some conceptually different strategies to simulate the fluid flow and traffic flow phenomena have been developed using the cellular automata techniques introduced by John von Neumann and Stanislaw Ulam in the early 1950s (von Neumann, 1951, 1966).

In the first case, fully discrete models obeying cellular automata rules have been worked out for the microscopic motion of the particles of a gas, such that the coarse-grained behaviour (in the thermodynamic limit) lies in the same universality class as the fluid flow phenomenon. This class of dynamical systems, now known as lattice gas models, consist of a regular lattice, each site of which can have a finite number of states representing the directions of motion of the gas particles, and evolves in discrete time steps obeying a set of homogeneous local rules which define the system dynamics. These rules are defined in such a way that the physical laws of conservation of mass, momentum and energy are fulfilled during the propagation and collisions of the gas particles (Boghosian, 1999). Typically, only the nearest neighbours are involved in the updating of any lattice site.

The first attempt along these lines was undertaken by Leo P. Kadanoff and Jack Swift in 1968 (Kadanoff & Swift, 1968). The Kadanoff-Swift model exhibits many features of real fluids, such as sound-wave propagation, and long-time tails in velocity autocorrelation functions. As the authors noted, however, it does not faithfully reproduce the correct motion of a viscous fluid (Boghosian, 1999). The next advance in the lattice modelling of fluids came in the mid 1970's, when J. Hardy, O. de Pazzis and Y. Pomeau introduced a new lattice model (the HPP model, named for its authors) with a number of innovations that warrant discussion here (Hardy et al., 1973, 1976). Like the model of Kadanoff and Swift, the HPP model gives rise to anisotropic hydrodynamic equations that are not invariant under a global spatial rotation. At the time, this was not considered a problem, since the real purpose of these models was to study the statistical physics of fluids, and both models could

do this well. Traditional computational fluid dynamicists, however, were not inclined to take notice of this as a serious numerical method unless and until a way was found to remove the unphysical anisotropy (Boghosian, 1999). Thirteen years passed from the introduction of the HPP model to the solution of the anisotropy problem in 1986 by Uriel Frisch, Brosl Hasslacher and Yves Pomeau (Frish et al., 1986), and simultaneously by Stephen Wolfram (Wolfram, 1986). Frisch, Hasslacher and Pomeau demonstrated that it is possible to simulate the Navier-Stokes fluid flows by using a cellular automata gas model on a hexagonal lattice, with extremely simple translation and collision rules governing the movement of the particles. In the FHP model, named after the authors of the first reference given above, all the particles have unit mass and move with the same speed hopping from site to site in a hexagonal two-dimensional lattice. The dynamics of this system involves a set of collision rules that conserve the number of particles and momentum (kinetic energy is trivially conserved). From a strict theoretical point of view, it is not clear at the present time if the lattice gas collective equations are equivalent to the Navier-Stokes equations, or if they include them as a particular case. However, there has been a growing interest in studying the viscous fluid flow using lattice gas models due to its great facility to handle complex boundary and initial conditions, and also because the computer simulations have shown that lattice gases behave like normal fluids under some restricted conditions (Hasslacher, 1987; Salcido & Rechtman, 1991, 1993; Rechtman & Salcido, 1996; Salcido, 1993, 1994). The FHP model, in particular, is now considered as an efficient way to simulate viscous flows at moderate Mach numbers in situations involving complex boundaries. However, it is unable to represent thermal or diffusional effects since all particles have the same speed and are of the same nature (Chen et al., 1989). Maybe the simplest lattice gas with thermal properties is a nine-velocity model defined on a square two-dimensional lattice where particles may be at rest or travelling to their nearest or next nearest neighbours (Chen et al., 1989; Rechtman et al., 1990, 1992; Salcido & Rechtman, 1991, 1993; Rechtman & Salcido, 1996).

In the field of air pollution, one of the first attempts to use a cellular automata lattice gas approach for modelling transport and dispersion phenomena of air pollutants can be found in the work by A. Salcido (Salcido, 1993, 1994; Salcido et al., 1993). There, it is shown how the lattice gas rules, in spite of their relative simplicity, are sufficient to simulate, at least qualitatively, some complex processes affecting unsteady dispersion, including momentum exchange with the surrounding atmosphere and deposition. More recent attempts are found in the work by A. Sciarretta and R. Cipollone (Sciarretta & Cipollone, 2001, 2002; Sciarretta 2006), where a comprehensive stochastic lattice gas model, which provides also reliable quantitative predictions, is presented. Lattice gas approaches to the wind field estimation problem have been developed also (Salcido et al., 2008; Salcido & Celada, 2010).

Simultaneously with the development of the lattice gas models, a new class of microscopic traffic models emerged also within the conceptual framework of the cellular automata. These new models, known as cellular automata traffic models or traffic cellular automata, are dynamical systems that are discrete in nature, in the sense that the roads are represented by one-dimensional (1D) or two-dimensional lattices, each lattice site being empty or containing exactly one vehicle, and time advances with discrete steps. The first studies in this field were done by Cremer and Ludwig in 1986 (Cremer & Ludwig, 1986). They proposed a fast simulation model for traffic flow through urban networks. In their model, the progression of cars on a street was simulated by moving 1-bit variables through binary positions of bytes in the storage which were arranged to model the topology of a specified network. Also, in terms of some boolean operations, the model was enabled to perform

diverse movements of a vehicle, like driving at a constant speed, lane changing, passing, decelerating and accelerating, queueing and turning at intersections. Nevertheless, it was up to the first half of the nineteen nineties, with the proposals of Nagel and Schreckenberg in 1992 (Nagel & Schreckenberg, 1992) and of Fukui and Ishibashi in 1996 (Fukui & Ishibashi, 1996a), that cellular automata attracted attention as microscopic traffic models. From then on, traffic scientists have been carrying out many studies about the possibilities of using approaches of cellular automata for building models of traffic that not only are well-formulated from the view of physics and able to reproduce the main behavioural aspects of real vehicular traffic, but also being efficient and practical for computer implementation.

Although traffic cellular automata are quite similar to the cellular automata fluids in several respects, and one can talk about the system like a lattice gas in both cases, in contrast to the fluid models, the particles in a traffic model could be considered, or better yet, would have to be considered as intelligent objects, able to learn from past experience, thereby opening the door to the incorporation of behavioural and psychological aspects (Helbing, 2001; Maerivoet & De Moor, 2005).

In this chapter, we will not consider the full process of analysis of impact of traffic on urban air quality. Instead, we are interested only in that stage of analysis which is concerned with modelling the traffic flow for the purposes of estimating the distribution of the vehicles (mobile sources) in space and time. Specifically, we will be concerned just with a simple case of this problem, which deals with the simulation of the movement of identical vehicles, but at different speed, in a single lane highway. Within this framework, for example, we would like to be able of finding out the number density of the vehicles which are moving at any given point in the highway, at any given instant, for each speed possible value. So that, by means of an emission rate model, later we would be able to estimate also the distribution in space and time of the vehicular emissions of air pollutants. For these purposes, here we will consider in detail the analysis of the equilibrium properties of the 1D cellular automata traffic models, expecting to provide some general results about this class of traffic models that may contribute not only to improve the speed of the computer simulations, but also in advancing some steps towards a well-established theory of the traffic cellular automata.

In general, it is important to try to address these problems, or any others in this field, starting from the fundamental laws governing the traffic systems behaviour. Using theoretical approaches based on continuum or statistical physics, for more than a half a century physicists have been trying to understand the basic principles governing traffic phenomena and contributing to traffic science by developing models of traffic. The theoretical analysis and computer simulation of these models not only provide deep insight into the properties of the model but also help on improving understanding of complex phenomena observed in real traffic. Moreover, using these models, physicists have been calculating some quantities of interest in practical applications in traffic engineering (Chowdhury et al, 2000).

The rest of this chapter is organized as follows. In the next section, it is provided a general description of the main features and basic aspects of cellular automata and of the cellular automata models for traffic flow in a highway, including presentation and discussion of the main ones with some detail. In section 3, we presented and discussed the equilibrium theory of the cellular automata models for traffic flow in a single lane, and in the fourth section we provided a detailed comparison of the equilibrium properties of these models against the steady states of the Nagel-Schreckenger and Fukui-Ishibashi traffic cellular automata. Finally, a section devoted to conclusions and suggestions for future work was included.

2. Cellular automata and traffic flow models

At the suggestion of Stanislaw Ulam, cellular automata were introduced by John von Neumann in the early 1950s as very simple mathematical models to investigate self-organisation and self-reproduction (von Neumann, 1951, 1966). In contrast to the typical mathematical models of self-organisation such as dissipative nonlinear differential equations or iterated mappings, cellular automata provide an alternative approach, involving discrete coordinates and variables as well as discrete time. The main attractive feature of cellular automata is that in spite of their conceptual simplicity, which allows for an easiness of implementation for computer simulation, so as a detailed and complete mathematical analysis in principle, they are able to exhibit a wide variety of amazingly complex behaviour. Thus, numerous physical and other systems containing many discrete elements with local interactions, for example the dynamical Ising model, gas and fluid dynamics, traffic flow, various biological issues, growth of crystals, nonlinear chemical systems, and some many others, can be conveniently modelled as cellular automata (Toffoli, 1984; Doolen et al, 1990; Chopard & Droz, 1998; Wolfram, 1986b, 1994, 2002; Bagnoli, 2001; Stauffer, 2001, Maerivoet & De Moor, 2005).

2.1 Main features of cellular automata

In order to understand why and how cellular automata can be used as models for various systems in nature, we will begin by describing very briefly the main ingredients that constitute a cellular automaton: the physical environment, the states of the sites, their neighbourhoods, and finally a local transition rule. More complete and detailed descriptions can be found, for example, in the works of F. Bagnoli (Bagnoli, 2001) and B. Chopard and M. Droz (Chopard & Droz, 1998).

Cellular automata are fully discrete dynamical systems. The physical environment of a cellular automata system is constituted of a finite-dimensional lattice, with each site (cell or box) having a finite number of discrete states. The state of the system is completely specified by the states at each lattice site. It evolves in time in discrete steps, and its dynamics is specified by some fixed and definite rule of evolution, which may be deterministic or non-deterministic (probabilistic), and, in general, may have many simplifying features: it is homogeneous (all sites evolve by the same rule) although inhomogeneous cellular automata can been considered too; it is spatially local (the rules for the evolution of each site depend only on the state of the site itself and the states of sites in its local neighbourhood); it allows for synchronous updating (all cells can be updated simultaneously); it is temporally local (the rule depends only on cell values at the previous time-step, or a few previous ones). Figure 2 illustrates the most common neighbourhoods used with cellular automata defined on one and two-dimensional lattices.

For 2D (and higher dimensional) cellular automata, the number of nearest and next nearest lattice sites contained in the neighbourhood depends on the lattice topology. In Fig. 2, it was illustrated only the case of a square 2D lattice.

The entirely local construction of cellular automata has for a crucial consequence the fact that cellular automata rules define no intrinsic length scale other than the size of a single site and its neighbourhood, and no intrinsic time scale other than the duration of a single time step. In the infinite time limit the configurations are self-similar, and views of the configuration with different magnifications are indistinguishable.

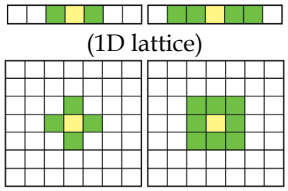


Fig. 2. Neighbourhoods commonly used with 1D and 2D cellular automata. The von Neumann neighbourhood (left) consists of the central site itself (in yellow) plus the nearest neighbours (in green), and the Moore neighbourhood (right) is composed of the central site itself (yellow) plus the nearest and next nearest neighbours (in green).

In practice, the computer simulations using cellular automata models are carried out on a finite rather than an infinite lattice, and therefore it is important to consider how to handle the sites on the edges, as this can affect the values of all the sites in the lattice. It is possible to define neighbourhoods differently for the sites on the boundary, but then new rules for them have to be defined as well. Another possibility is fixing the values of these sites to remain constant, corresponding to Dirichlet boundary condition for partial differential equations. Most often periodic boundary conditions are assumed, where the first and the last sites are identified; for instance, in one dimension the lattice sites are treated as if they lay on a circle of finite radius, and similarly for higher dimensions. This also solves the boundary problems with neighbourhoods.

Despite their conceptual simplicity, cellular automata are capable of diverse and complex behaviour. For most cellular automata models, however, the only general method to determine the qualitative (average) dynamics of the system is to run simulations on a computer for several initial global configurations. Cellular automata classification based on the study of its dynamics has been a major focus for the researchers. Simulations suggest that the patterns generated in the time evolution of cellular automata from disordered initial states can be classified as follows (Wolfram, 1984): Class 1: cellular automata which evolve to a homogeneous state; Class 2: displaying simple separated periodic structures; Class 3: which exhibit chaotic or pseudo-random behaviour; and Class 4: which yield complex patterns of localized structures and are capable of universal computation.

2.2 Traffic cellular automata

The application of cellular automata to traffic dynamics goes back to Cremer and coworkers (Cremer and Ludwig, 1986; Schütt, 1991), to Nagel and Schreckenberg (Nagel & Schreckenberg, 1992), and to Fukui and Ishibashi (Fukui & Ishibashi, 1996a). Other early cellular automaton studies were carried out by Biham et al. (Biham et al., 1992). These proposals of microscopic traffic models stimulated an enormous amount of research activity, aimed at understanding and controlling traffic instabilities, which are responsible for stop-and-go traffic and congestion, both on freeways (Sasvári & Kertész, 1997) and in cities (Helbing, 2001). Since then, cellular automata became popular for the microscopic simulation of traffic flow (Vilar & de Souza, 1994; Chowdhury et al, 2000; Helbing, 2001; Nagatani, 2002; Nagel et al., 2003; Maerivoet & De Moor, 2005), including multilane highways (Wagner et al., 1997; Nagel et al., 1998; Chowdhury et al., 2000; Nagel et al., 2003; Maerivoet & De Moor, 2005) and complex urban traffic networks (Fukui & Ishibashi, 1996b; Fukui et al., 1996; Esser & Schreckenberg, 1997; Rickert & Nagel, 1997; Nagel & Barrett, 1997;

Simon & Nagel, 1998; Maerivoet & De Moor, 2005). Nowadays, there exist an overwhelming number of proposals and publications in this field.
Here, however, we will focus our interests on the cellular automata models for unidirectional single-lane traffic flow with periodic boundary conditions. Some insight to the importance of studying this basic problem can be obtained by considering, for example, traffic flows on unidirectional two-lane motorways: Drivers, in many countries, are by law obliged to drive on the right hand lane, unless when performing overtaking manoeuvres. A frequently observed phenomenon is then that under light traffic conditions, a slower moving vehicle is located on the right lane, and is acting as a moving bottleneck. As a result, all faster vehicles will line up on the left lane (overtaking on the right lane is prohibited by law), thereby causing a population inversion in the lanes. It is under these circumstances that the stability of the car-following behaviour plays an important role (Maerivoet & De Moor, 2005). Even for multi-lane traffic, its dynamics is essentially that of parallel single lanes when considering densely congested traffic flows. Studying these simplified traffic flow conditions is, in fact, the easiest way to determine whether or not internal effects of a traffic flow model play a role in, for example, the spontaneous breakdown of traffic, as all external effects (i.e., the boundary conditions) are eliminated (Nagel & Nelson, 2005). Nevertheless, when applying these models to real-life traffic networks, closed-loop traffic is not very representative, as the behaviour near bottlenecks plays a far more important role (Helbing, 2001).

2.2.1 Common features in cellular automata models for traffic flow in a single-lane

For the basic problem of traffic flow of identical vehicles in a single-lane, there are three cellular automata models that we consider important for our purposes in this work: the model defined by the Wolfram's rule CA-184, and the original models proposed by Nagel and Schreckenberg (Nagel & Schreckenberg, 1992) and by Fukui and Ishibashi (Fukui & Ishibashi, 1996a). These models (hereafter referred as WR184, NS and FI, respectively), so as most of cellular automata models for unidirectional single-lane traffic flow, have the following basic common characteristics: each of them can be considered as a 1D lattice gas of undistinguishable particles with unit mass (model cars) which obey an exclusion principle (no more than one particle may occupy any lattice site at any time), can be at rest or moving with positive integer velocities v up to an upper limit v_{max} (reverse motion is forbidden and there exists an speed limit), and interact each other according to a specific set of parallel updating-rules (applied synchronously to all particles) that conserve the number of particles and prevent collisions (car crashes) and overtaking, but do not conserve momentum and energy of the particles. The main difference between these models is concerned with the particular procedure that is implemented to change the speed of the lattice gas particles. In the next three subsections we describe the main features of the sets of rules (updating rules) of the models WR184, NS and FI that are consecutively applied to all vehicles in the lattice.

2.2.2 The Wolfram´s CA184 traffic model

The simplest one-dimensional cellular automata model for highway traffic flow is the model defined by the Wolfram´s rule CA-184. This is a deterministic cellular automata model whose dynamics is defined by the following two rules:

R1. Acceleration and braking: $v_i(t+1) \leftarrow \min\{ h_i(t), 1 \}$
R2. Vehicle movement: $x_i(t+1) \leftarrow x_i(t) + v_i(t+1)$

Rule R1 sets the speed v_i of the i-th vehicle, for the current updated configuration of the system; it states that a vehicle always strives to drive at a speed of one lattice site per timestep, unless its impeded by its direct leader, in which case $h_i(t)$, the number of empty sites in front of the i-th vehicle at time t, is equal zero, and the vehicle consequently stops in order to avoid a collision. The rule R2 just allows the vehicles to advance in the lattice.
The Wolfram's rule 184 can be expressed also as follows. The state of each lattice site at any time is expressed by a 1-digit binary number, whose value is *1* if the site is occupied by a particle and *0* otherwise. For any lattice site, i, the state at time $t+1$, denoted by $\sigma(i, t+1)$, will be a function of the states $\sigma(i-1, t)$, $\sigma(i, t)$, and $\sigma(i+1, t)$, at time t, in the sites which compose the Moore neighbourhood of the site in question, $\mathcal{N}_i = \{i-1, i, i+1\}$. The configuration of the states of the sites in $\mathcal{N}_i$ is expressed as a 3-digit binary number $\xi(i,t) = \sigma(i-1,t)\sigma(i,t)\sigma(i+1,t)$. Then the evolution in time of the state at the lattice site i can be written as

$$\sigma(i, t+1) = \mathcal{F}(\xi(i,t))$$

where the function $\mathcal{F}$ is defined by the updating rule given in Table 1.

ξ(i,t)	111	110	101	100	011	010	001	000
σ(i, t+1)	1	0	1	1	1	0	0	0

Table 1. Wolfram's rule 184. All eight possible configurations for the local neighbourhood are sorted in the first row, and the results are shown in the second row. The physical meaning is that a particle (a *1*) moves to the right if its right neighbouring site is empty.

In Fig. 3, we show the evolution in time of the traffic model WR184. We considered a lattice consisting of *500* sites with periodic boundary conditions, and carried out simulations over a period of *465* timesteps each, for mean densities $n = 0.15, 0.25, 0.35, 0.45, 0.5, 0.55, 0.65, 0.75,$ and *0.85* particles/site. Each case, the initial condition was prepared by distributing the particles randomly in the lattice.
In the figures, the time and space axes are oriented from left to right, and top to bottom, respectively. The simulations show the occurrence of a free-flow regime for low densities (first row); a transition from a free-flow to a congested-flow regime for densities around the critical density $n_c = 0.50$ particles/site (second row); and a congested-flow regime for high densities (third row). As time advances, the congestion waves can be seen propagating in the opposite direction of traffic. We can see also that the WR184 model constitutes a fully deterministic system that continuously repeats itself. A characteristic of the encountered congestion waves is that they have an eternal life time.
Let $n_0(t)$ and $n_1(t)$ denote the average numbers of particles per lattice site at time t, which are at rest ($c_0 = 0$) and moving with the speed one ($c_1 = 1$), respectively. Then the mean flow is given by $q = n_0c_0 + n_1c_1 = n_1$, and the mean speed is $v = q/n = n_1/n$, where $n = n_0 + n_1$ is the mean density of particles in the lattice. In Fig. 4, there are shown the plots of n_0, n_1, q and v as functions of n for the steady state of the WR184 model. As can be seen from the plot drawn in green, the mean speed remains constant at $v = c_1 = 1$ sites per timestep, until the critical density $n_c = 0.5$ particles/site is reached, at which point v will start to diminish towards zero where the density $n = 1$ particles/site is reached. Similarly, the mean flow q (plot drawn in red) first increases and then decreases linearly with the density, below and respectively above, the critical density. Here, the capacity flow $q_{cap} = 0.5$ particles/timestep is reached. The transition from the free-flowing to the congested regime is characterised by a

population inversion from the particles in motion (with density n_1; plot drawn in red) to the particles at rest (with density n_0; plot drawn in blue). As is evidenced by the isosceles triangular shape of the fundamental diagram (q as function of n) of the WR184 traffic model, there are only two possible kinematic wave speeds: $c_w = \pm 1$ site/timestep. Both speeds are also clearly visible in the first row, respectively third row, time–space diagrams of Fig. 3.

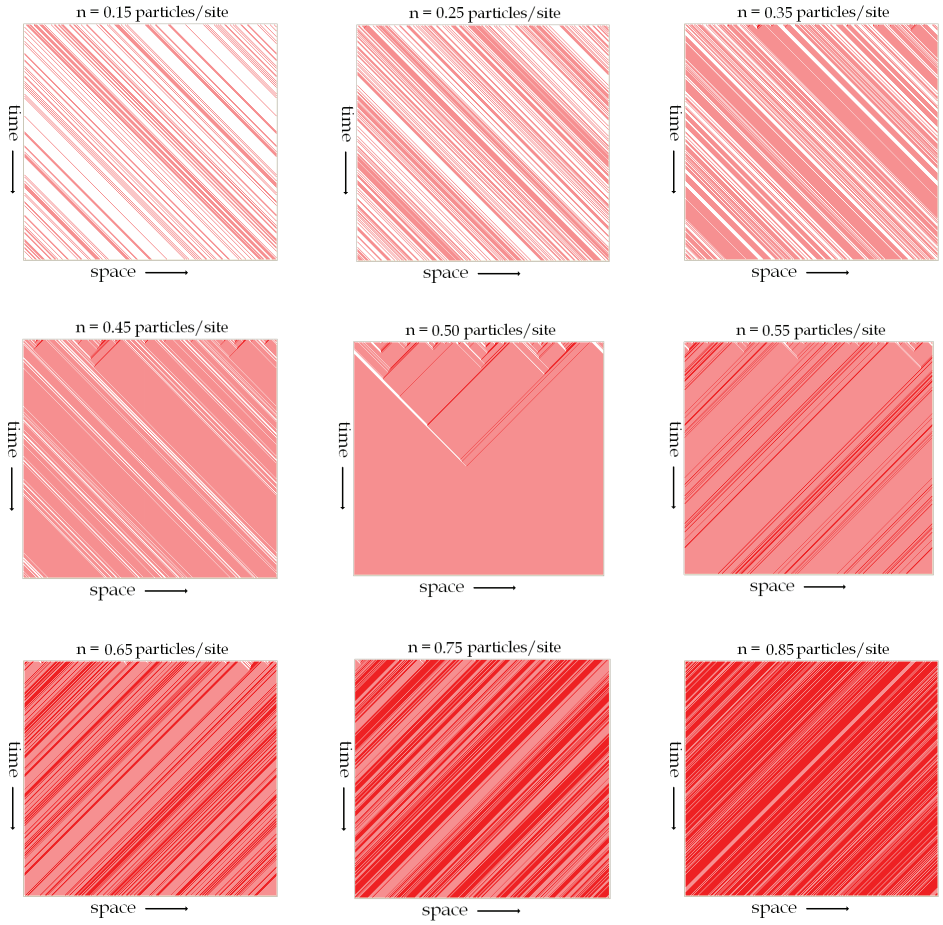


Fig. 3. Typical time–space diagrams of the WR184 traffic model. The shown ring-geometry lattices each contain *500* sites, with a visible period of *465* timesteps (each vehicle is represented as a single coloured dot). First row: vehicles driving a free-flow regime with mean densities $n = 0.15, 0.25$ and 0.35 particles/site. Second row: a transition from the free-flow regime to the congested one, occurring for densities around $n = 0.50$ particles/site. Third row: vehicles driving in a congested regime with $n = 0.65, 0.75$ and 0.85 particles/site. The congestion waves can be seen as propagating in the opposite direction of traffic.

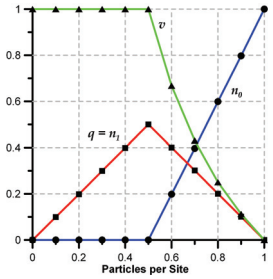


Fig. 4. Typical behaviour diagrams of the WR184 model, based on global measurements on the lattice carried out at the steady state. Green (▲): mean speed remains constant at $v = 1$ site/timestep, until the critical density $n_c = 0.5$ is reached, at which point v will start to diminish towards zero. Red (■): flow diagram, with its characteristic isosceles triangular shape. The transition between the free-flowing and the congested regimes is observed. Blue (●): the number of particles at rest remains null ($n_0 = 0$) until the critical density is reached, at which point it starts to increase towards one ($n_0 = 1$). The transition between the free-flowing and the congested regimes is close related to a population inversion between moving particles and particles at rest (plots identified by symbols (■) and (●), respectively).

2.2.3 The Nagel-Schreckenberg traffic model

In 1992, Kai Nagel and Michael Schreckenberg proposed a very simple stochastic cellular automata traffic model (Nagel & Schreckenberg, 1992). In the NS model, space and time are discrete and hence also the velocities. The road, which is supposed unidirectional, is modelled by a 1D lattice with L sites (cells or boxes) that represent the positions of the vehicles. The number of sites in the lattice may be considered finite or infinite. The distance between adjacent lattice sites is defined as unit in this work, although it is often determined by the front-bumper to front-bumper distance of cars in the densest jam and is usually taken to be 7.5 m. Each site can either be empty or occupied by one, and only one particle (car or vehicle), which can be at rest ($v = 0$) or moving along the lattice (always in the same direction, hereafter assumed from left to right) with a integer speed $v = 1, 2, 3, \ldots, v_{max}$. The evolution of the system in time (its dynamics) is defined by the following four rules, which must be applied to all particles (i.e. to all the non-empty lattice sites) simultaneously (Nagel & Schreckenberg, 1992). If at time t, there is a particle at site k ($k = 1, 2, 3, \ldots, L$), then

R1. **Acceleration:** the particle´s speed $v(k, t)$ is substituted by the smallest of $v(k, t) + 1$ and v_{max}. That is: $v(k, t) \rightarrow u(k, t) = \min\{ v(k, t) + 1, v_{max} \}$

R2. **Braking:** if $d(k,t)$, the number of the empty sites ahead the particle at time t, is smaller than $u(k, t)$, then $u(k, t) \rightarrow w(k, t) = \min\{ d(k, t), u(k, t) \}$

R3. **Randomization**: with probability p, the speed of the particle at time $t+1$ is set equal to the largest of $w(k, t) - 1$ and 0. That is: $v(k, t+1) = \max\{ w(k, t) - 1, 0 \}$

R4. **Driving:** the particle moves hopping from site k to site $k + v(k, t+1)$.

The number of empty sites in front of a car is called headway. For $v_{max} = 5$ a calibration of the model showed that each timestep $t \rightarrow t + 1$ corresponds to approximately 1 sec in real time (Nagel & Schreckenberg, 1992). Hereafter we will consider only a lattice with periodic boundary conditions, so that the number of particles is conserved. The maximum velocity v_{max} can be interpreted as a speed limit that drivers are obligated to respect, and therefore it will be taken to be identical for all particles. Fig. 5 shows a typical configuration.

Fig. 5. A possible configuration during the time evolution of the Nagel and Schreckenberg traffic model. The lattices sites have been drawn with different colors for evidencing the speeds of the vehicles.

The four updating rules of the NS model have simple interpretations within the traffic jargon context (Schadschneider, 1999). The first rule (R1) means that drivers want to drive as fast as allowed. The rule R2 means that drivers have to brake to avoid collision with the vehicle ahead. The rule R3 (randomization) takes into account several effects, e.g. road conditions (e.g. slope, weather) or psychological effects (e.g. velocity fluctuations in free traffic). An important consequence of this rule is the introduction of overreactions at braking which are crucial for the occurrence of spontaneous jam formation (Schadschneider, 1999). Although this rationale is widely agreed upon, much criticism was however expressed due to the rule R3. In particular, Brilon and Wu believe that this rule has no theoretical background and is in fact introduced quite heuristically (Brilon & Wu, 1999). The last rule (R4) implements the displacement of the vehicles. Thus the NS model captures the features of gradual acceleration, deceleration and randomization in realistic traffic flows and, in agreement with the results of the computer simulations, it seems that all four rules, R1-R4, are necessary to reproduce the basic properties of real traffic; therefore this model is considered as a minimal model. An intuitive feeling for the NS model dynamics can be obtained from the nine time–space diagrams presented in Fig. 6.

The diagrams in Fig. 6 were obtained as follows. We considered a lattice consisting of *500* sites with periodic boundary conditions, and carried out simulations over a period of *465* timesteps each. In the figure, we have arranged these diagrams in a 3x3 matrix, for illustrating several aspects of the evolution of the NS traffic model with $v_{max} = 1$. The matrix rows correspond to mean densities $n = 0.25, 0.50$ and 0.75 particles per site, and the columns correspond to randomization probabilities $p = 0.25, 0.50$ and 0.75. In the figures, the time and space axes are oriented from left to right, and top to bottom, respectively. As can be seen in the diagrams, the randomization rule (R3) gives rise to many unstable artificial phantom mini-jams. The downstream fronts of these jams smear out, forming unstable interfaces (Nagel et al., 2003). This is a direct result of the fact that the intrinsic noise (as embodied by p) in the NS model is too strong: a jam can always form at any density, meaning that breakdown will occur, even in the free-flow traffic regime. For low enough densities however, these jams can vanish as they are absorbed by vehicles with sufficient space headways or by new jams in the system (Krauß, et al., 1999).

In Fig. 7, for the NS model with $v_{max} = 1$, and in Fig. 8, for $v_{max} = 2$ (top row) and $v_{max} = 3$ (bottom row), there are shown steady-state simulation results for the mean flow q and the partial densities n_v (global average numbers of the particles per site which move with the speed v) as functions of n, for randomization probabilities $p = 0.0, 0.25$ and 0.50. The simulations were carried out using a *860*-sites lattice with periodic boundary conditions, and proceeding as follows: for density values increasing from *0* to *1* with steps of $\Delta n = 0.01$, the system was allowed to evolve for *1000* timesteps, and each simulation run was repeated *20* times. As can be seen in Fig. 7, although the NS model with $v_{max} = 1$ and $p = 0$ has exactly the same behaviour as the WR184 model, important and growing deviations from this model become evident as p increases from zero. The top and bottom rows of Fig. 8 show the steady state behaviour of the NS model for $v_{max} = 2$ and $v_{max} = 3$, respectively, for three

values of the parameter p. In both cases, when $p = 0$ the system remains under the free-flowing regime (all the particles moving with the maximum speed) until the mean density n, growing from zero, reaches the critical densities $n_c = 1/3$ and $n_c = 1/4$, respectively.

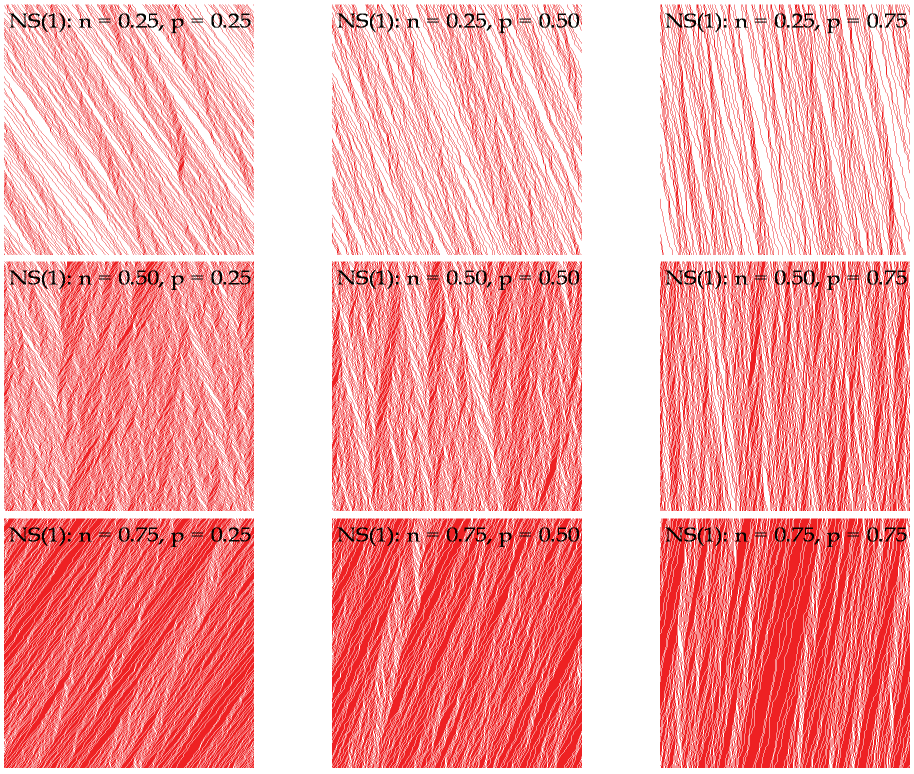


Fig. 6. Time-space diagrams showing the behaviour of the Nagel and Scherckenberg traffic model for several values of density n and randomization parameter p. Simulations were carried out on a 500 sites lattice with periodic boundary conditions, for periods of 465 timesteps.

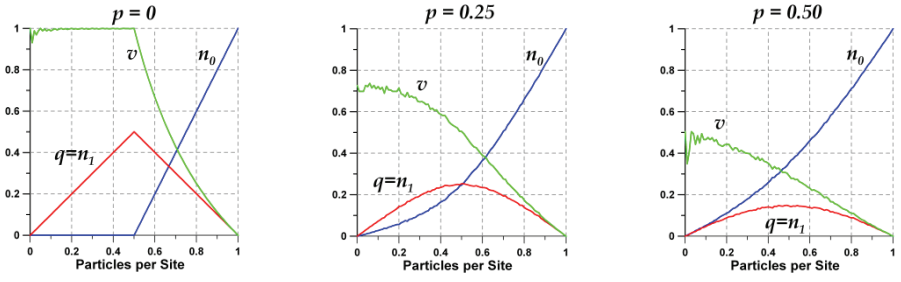


Fig. 7. Steady-state behaviour of the NS model with $v_{max} = 1$. The diagrams show the effect of the randomization p on the mean speed v and the partial densities n_v as functions of density n. This model with $p = 0$ is exactly the same as the WR184 model.

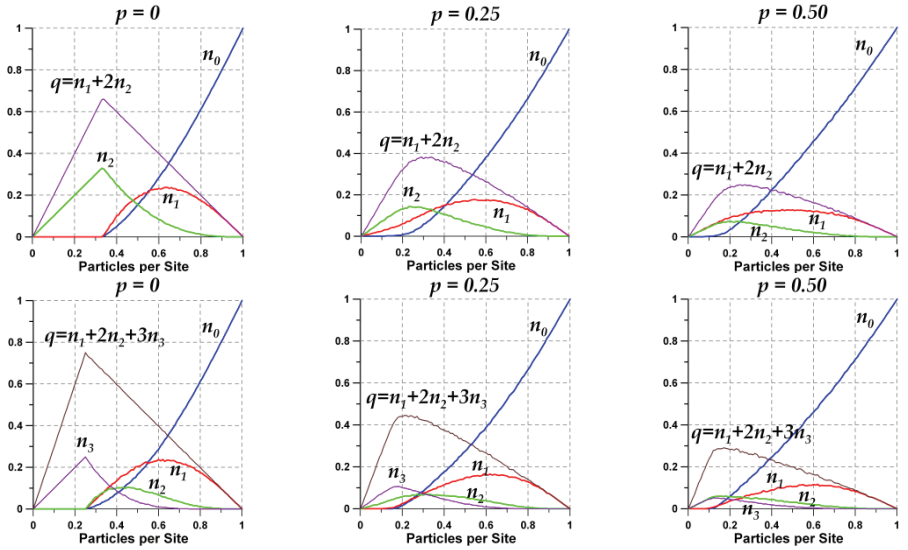


Fig. 8. Steady states of the NS model with $v_{max} = 2$ (top row) and $v_{max} = 3$ (bottom row) for randomization probabilities $p = 0.00, 0.25$ and 0.50. For $p = 0$ the system remains under a free-flowing regime until density n reaches the values $n_c = 1/3$ and $n_c = 1/4$, respectively.

2.2.4 The Fukui-Ishibashi traffic model

In 1996, M. Fukui and Y. Ishibashi (Fukui & Ishibashi, 1993, 1996a; Wang et al., 1997) proposed another cellular automata model for traffic flow in a single lane (hereafter referred as FI), where the cars can move by at most v_{max} lattice sites in one timestep if they are not blocked by cars in front. In detail, if the number of empty sites h in front of a car is larger than v_{max} at time t, then it can move forward v_{max} (or $v_{max} - 1$) sites in the next time-step with probability $1 - f$ (or f). Here, the probability f represents the degree of stochastic delay. From the point of view of this model, no driver would like to slow down when far away from the vehicle ahead. In the high density case, the stochastic delay in this model represents the assurance of the avoidance of crashes. The model with $f = 0$ is referred to as the deterministic FI model with maximum speed v_{max}, while the case with $f = 1$ is the deterministic FI model with maximum speed $v_{max} - 1$. If $h < v_{max}$ at time t, then the car can only move by h sites in the next time-step. The FI model differs from the NS model in that the increase in speed may not be gradual, and that stochastic delay only applies to the high speed cars.

In Fig. 9, the steady state behaviour of partial densities $n_k(n)$ and traffic flow $q(n)$ is shown for the FI models with $v_{max} = 1$ (first row) and $v_{max} = 2$ (second row) for stochastic delay values $p = 0.00, 0.25$ and 0.50. The diagrams were obtained by means of computer simulations carried out using a 860-sites lattice with periodic boundary conditions. For density values increasing from zero to 1 with steps of $\Delta n = 0.01$, the system was allowed to evolve for 1000 timesteps, and each simulation run was repeated 20 times. The results showed that, in both cases, $v_{max} = 1$ and $v_{max} = 2$, when $p = 0$ the system remained under a free-flowing regime (all the particles moving with the maximum speed) until density n, growing from zero, reached the critical densities $n_c = 1/2$ and $n_c = 1/3$, respectively. The results in the top row of Fig. 9 show that FI and NS models are equivalent to each other

when $v_{max} = 1$, independently of p; and they both are equivalent to WR184 model for $p = 0$. For the models with $v_{max} = 2$, comparison of the top row of Fig. 8 with the second row of Fig. 9 show important differences between the respective simulations with the models FI and NS. In the limit $p = 0$, the behaviour of the partial densities of the FI model as functions of the global density n is quite similar, although different, to the respective behaviour of the partial densities of the NS model. However, for $p > 0$ these models behave quite different from each other. For example, while in the NS model all the partial densities n_0, n_1 and n_2 are, in general, greater than zero for any density value $0 < n < 1$, in the FI model $n_0 = 0$, $n_1 > 0$ and $n_2 > 0$ for $0 < n < ½$, but $n_0 > 0$, $n_1 > 0$ and $n_2 = 0$ when $½ < n < 1$. As it is observed in Fig. 9, the FI traffic model (with $v_{max} = 2$ and $p > 0$) switches between two different two-speed models at $n = ½$: { $n_0 = 0$, $n_1 > 0$, $n_2 > 0$ } ↔ { $n_0 > 0$, $n_1 > 0$, $n_2 = 0$ }.

Finally, we mention the related work of Wang et al. who studied the stochastic model of Fukui and Ishibashi both analytically and numerically, providing an exact result for $p = 0$, and a close approximation for the model with $p \neq 0$ (Wang et al., 1998a). Based on the FI model, they developed a model that is subtly different. They assumed that drivers do not suffer from concentration lapses at high speeds, but are instead only subjected to the random deceleration when they are driving close enough to their direct frontal leaders (Wang et al., 2001). More recently, Lee et al. incorporated anticipation with respect to a vehicle's changing space gap as its leader is driving away. This results in a higher capacity flow, as well the appearance of a synchronised traffic regime, in which vehicles have a lower speed, but are all moving (Lee et al., 2002).

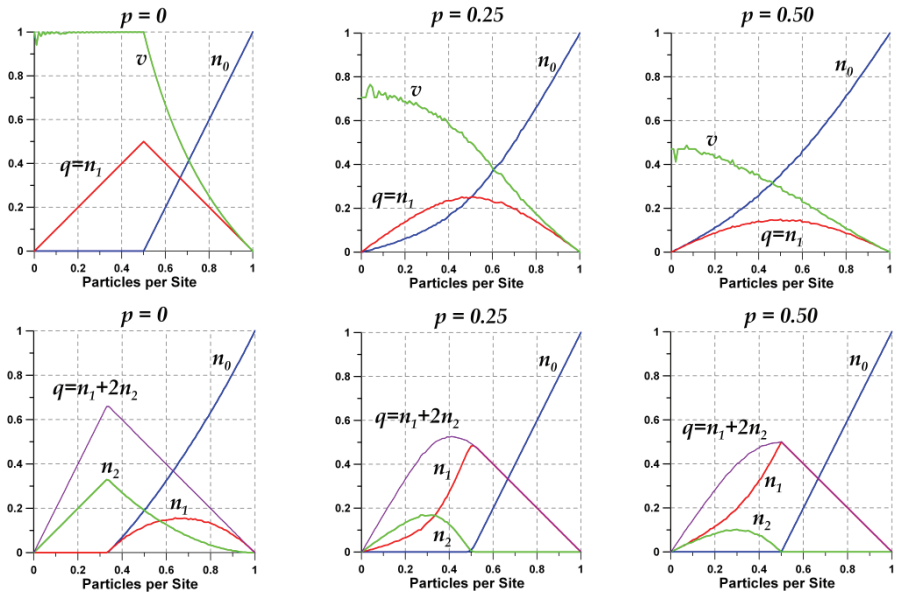


Fig. 9. The steady states of the FI models with $v_{max} = 1$ (top row) and $v_{max} = 2$ (bottom row) for stochastic delay values $p = 0.00, 0.25, 0.50$ and 0.75. The effect of the parameter p on the traffic flow q, the mean speed v, and the partial densities n_k, as functions of n, is illustrated.

3. Equilibrium properties of the 1D traffic cellular automata

As we have mentioned in Section 2, in the formulation of cellular automata traffic models, as it is the case of the WR184, NS and FI models, the interaction of the particles with each other is defined through some dynamical rules (deterministic and/or stochastic) which do not conserve the momentum and energy, and may drive the system far from equilibrium. The NS and FI models, in fact, have been considered as variants of the well-known asymmetric exclusion process (ASEP), the paradigm of the non-equilibrium systems (Schütz, 2001). As a consequence, notwithstanding their conceptual simplicity and easy construction, the analysis of the dynamics of a cellular automata traffic model is notoriously difficult in general. Big efforts are being made trying to apply the methods of non-equilibrium statistical physics to these systems, but only very few exact results have been obtained up to now. For the case of the NS model, the steady-state exact solution is known only if $v_{max} = 1$ (Schreckenberg et al., 1995; Evans et al., 1999). When $v_{max} > 1$, however, only approximations exist, and most of the existing results have been found through computer simulations (Schreckenberg et al., 1995; Nagel, 1996; Schadschneider & Schreckenberg, 1993, 1997). In the case of the Fukui-Ishibashi model, H. Fuks has derived an expression of the average car flow as a function of time (Fuks, 1999). For the same model, Boccara has studied a variational principle and its existence for other deterministic cellular automata models of traffic flow (Boccara, 2001). More recently, Wang et al. studied the non-deterministic FI model with arbitrary speed limit and degree of stochastic delay deriving a general expression for the average car speed in the steady state, which was found in excellent agreement with numerical data (Wang et al., 1998b). Furthermore, in the deterministic setting, many of the results are still on the "physical" level. In particular, they were not able to prove the convergence to the average velocity described by the fundamental diagram starting from any initial configuration of a given particle density even for the finite system, not speaking about infinite ones defined on the integer lattice (Blank, 2005, 2008). Another also open problem is the existence of invariant measures with a given particle density in the random setting with jumps greater than 1 (Blank, 2005, 2008). Some excellent reviews have been published in the last decade concerning the state of the art of traffic cellular automata theory (Chowdhury et al., 2000; Helbing, 2001; Nagatani, 2002; Nagel et al., 2003; Maerivoet & De Moor, 2005), however, up to the author´s knowledge, no study was reported about the equilibrium properties of the vehicle lattice gas prior to the paper published by Salcido in 2007 (Salcido, 2007).

If we know what the equilibrium states of a system are, then we can certainly know when this system is out of equilibrium, but this may not be true in reverse sense. In the theories of thermodynamics and statistical thermodynamics, once the entropy function of the system is known, the equilibrium states of the system can be defined as all those states which maximize entropy under certain conditions. For the NS and FI models, however, detailed balance condition is not obeyed (which, otherwise, is a condition for the system can be in thermodynamical equilibrium) and so, ordinary statistical mechanics is not applicable to study them. This is what we mean when saying that the rules defining NS and FI models continuously are driving the system out of equilibrium, and one can never see relaxation towards equilibrium states. But, if we introduce constraints that prevent a system of reaching equilibrium states in practice, it does not mean, at all, that the system has no equilibrium states in theory (or better said that one cannot define equilibrium states for it).

In the rest of this chapter, we will be considering a generic class of one-dimensional cellular automata models for multi-speed traffic flow with periodic boundary conditions (hereafter

referred as GC-1DTCA). We will assume that each model in this class has all the common basic features we described in Section 2.2.1, but no particular neither explicit specification of the dynamical updating rules of the model will be done. About these rules, we just will assume that they conserve the number of particles and that prevent collisions and overtaking by assigning the speed v to a particle if, and only if, it has, at least, a number v of free sites ahead. Within this framework, as we will see, an entropy function can be found for the models belonging to GC-1DTCA, which allows the study the properties of the equilibrium states (which here will be understood as the maximum entropy states) of the cellular automata models for multi-speed traffic flow in a single-lane.
After description of the model system and of the variables that will describe its state, as well as the identification of the microcanonical entropy function, the maximum entropy principle will be applied to determine the equilibrium state partial densities and the thermodynamic properties of the system, such as temperature, pressure, specific heat, and isothermal compressibility. The theoretical partial densities of the allowed velocities and fundamental diagrams will be compared with computer simulation results we obtained with the Nagel-Schreckenberg and Fukui-Ishibashi probabilistic cellular automata traffic models. In particular, as a part of this comparison, it is shown that, although the NS and FI traffic models behave as non-equilibrium systems, they evolve rapidly towards steady states (at least under periodic boundary conditions) which we have found very close to equilibrium under the view of our theoretical framework.

3.1 Entropy and maximum entropy states of 1D traffic cellular automata

Our system is a traffic cellular automaton defined on a 1D-lattice with L sites. It is assumed to have all the basic features cited in Section 2.2.1, but no particular or explicit specification of the velocity updating-rules is made here. However, concerning to these rules, we assumed they conserve the number of particles, and prevent collisions and overtaking by assigning the speed v to a particle if, and only if, it has, at least, a number v of free sites ahead. This means, in particular, that velocity anticipation is not considered here.
With this background, a particle with speed v (= *0, 1, .. v_{max}*) can be imagined as a brick of length $v + 1$ which has to be inserted in a 1D ring lattice (the brick row under question of a ring wall). This way, at any time t, the model system can be considered as one row of a ring wall, made of holes and $v_{max} + 1$ types of bricks (different in length) not overlapping each other. Since the dynamical rules of a particular model may change the lengths of the bricks, under certain conditions the system could reach states where the concentration of a particular type of bricks predominates over the others. The critical density is an upper bound of the density values for which only particles with speeds up to v (bricks with length $v + 1$) can be found in the system.

$$n_c(v) = \frac{1}{v+1} \tag{1}$$

The macroscopic state of the system will be described by the set of velocity distribution functions N_v (v = *0, 1, …, v_{max}*), each defined as the number of particles with some speed v in the lattice. The intensive variables defined as $n_v = N_v/L$ are global partial densities of the system. Then, the global density of the number of particles, $n = N/L$, the traffic flow (or momentum per site), q, and the kinetic energy per site, ε, of the system, are defined as

$$\sum_{v=0}^{v_{max}} n_v = n, \quad \sum_{v=0}^{v_{max}} v n_v = q, \quad \sum_{v=0}^{v_{max}} \varepsilon_v n_v = \varepsilon \tag{2}$$

Here ε_v stands for the kinetic energy of a particle with unit mass and speed v. It is easy to show that flow q and kinetic energy ε of the system cannot exceed, respectively, the maximum values $q_{max}(n)$ and $\varepsilon_{max}(n)$ given by

$$q_{max}(n) = \begin{cases} n\, v_{max} & 0 \le n \le n_c(v_{max}) \\ (1-n) & n_c(v_{max}) \le n \le 1 \end{cases} \tag{3}$$

$$\varepsilon_{max}(n) = \begin{cases} \frac{1}{2} n v_{max}^2 & 0 \le n \le n_c(v_{max}) \\ \frac{1}{2}(1-n) v_{max} & n_c(v_{max}) \le n \le 1 \end{cases} \tag{4}$$

in the thermodynamic limit $L \to \infty$, where $n_c(v_{max})$ is the critical density for $v = v_{max}$. Diagrams of $q_{max}(n)$ and $\varepsilon_{max}(n)$ are shown in Fig. 10. In the free-flow regime, $n < n_c(v_{max})$, vehicles move with speed v_{max}, and the gap between vehicles is either v_{max} or larger. In consequence, the traffic flow in this regime is $q_{max} = n v_{max}$. If global density is larger than the critical density, i.e. $n > n_c(v_{max})$, only $(L\text{-}N)/v_{max}$ vehicles can move with maximum speed and, in the limit $L \to \infty$, the maximum traffic flow is given $q_{max} = 1 - n$. Similarly, it can be obtained the maximum value ε_{max} of kinetic energy for both $n < n_c(v_{max})$ and $n > n_c(v_{max})$. As a consequence, the possible macroscopic states of the system are defined by those partial densities n_v which correspond to values of particle densities n and kinetic energies ε, ranging in the intervals $0 \le n \le 1$ and $0 \le \varepsilon \le \varepsilon_{max}(n)$, respectively.

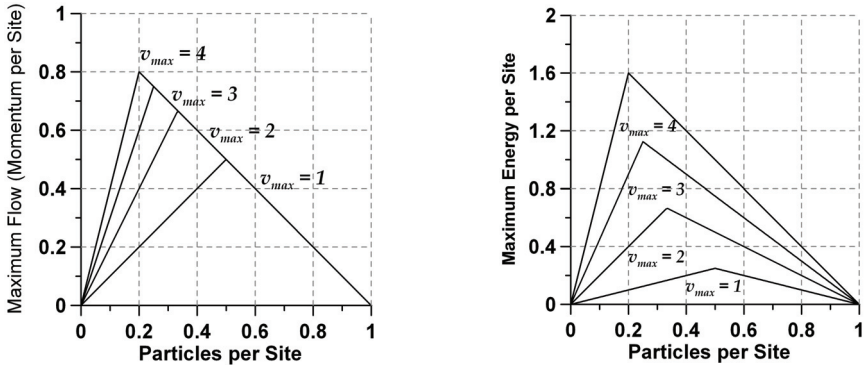


Fig. 10. Maximum flow q_{max} (left) and maximum energy ε_{max} (right) as functions of the density of particles n, for models with $v_{max} = 1, 2, 3$ and 4. For each $n \in [0, 1]$, the possible states of the system are those with energy $\varepsilon \in [0, \varepsilon_{max}(n)]$. The transition points correspond to the critical densities $n_c = 1/2, 1/3, 1/4$ and $1/5$.

Given initial and boundary conditions, the specific dynamical rules of the considered traffic cellular automata will define the macroscopic state of the system at any time t. Macroscopically, the state of the system will be characterized by the set of values of its partial densities n_v, or by the numbers $N_v = n_v L$, which is equivalent. Microscopically, however, there are many different arrangements in the lattice of given numbers (N_0 , N_1, ... ,

N_{vmax}) of particles moving there with speeds (*0, 1, 2, ..., v_{max}*), respectively, including a number Λ of sites which must remain empty. For models belonging to the GC-1DTCA, with periodic boundary conditions, the number $\Omega(L, N_v)$ of all these different microscopic arrangements of moving particles is given by

$$\Omega = \left(\frac{L}{\Lambda + N} \right) \left(\frac{(\Lambda + N)!}{\Lambda! N_0! N_1! \cdots N_{v_{max}}!} \right) \tag{5}$$

and the number of empty sites in the lattice (i.e., the lattice sites available for speeding up the particles) can be expressed as

$$\Lambda = L - \sum_{v=0}^{v_{max}} (v+1) N_v \geq 0 \tag{6}$$

Negative values of Λ are forbidden because of the non-anticipation restriction that we imposed to the models in the class we are considering.

We underline that $\Omega(L, N_v)$ is the number of all the possible configurations in which we can arrange N_0 bricks with *1*-site length (representing the particles at rest), N_1 bricks with *2*-sites length (representing the particles moving with speed v = *1*), N_2 bricks with *3*-sites length (representing the particles moving with speed v = *2*), and so on, up to N_{vmax} bricks with (v_{max}+*1*)-sites length (particles with speed v_{max}), by inserting them with no overlaps in a 1D ring-shaped lattice with L sites in total, but allowing that a number Λ of sites remain empty. We underline also that this result is valid for any cellular automata traffic model with all the common features we have specified above (Section 2.2.1). However, since particular dynamical rules of a model could prevent the system of reaching some microscopic states with particular configurations of particles, Eqn. (5) will provide, at least, for such cases, an upper bound to the number of them for that model.

Starting from $\Omega(L, N_v)$, the entropy function is defined by $S = ln(\Omega)$. Then, with the help of Stirling's approximation, the entropy per site, $s = S/L$, in the thermodynamic limit ($L \to \infty$), can be expressed as

$$s = (\lambda + n)\ln(\lambda + n) - \lambda \ln \lambda - \sum_{v=0}^{v_{max}} n_v \ln n_v \tag{7}$$

where

$$\lambda = \frac{\Lambda}{L} = 1 - \sum_{v=0}^{v_{max}} (v+1) n_v \tag{8}$$

As a pretty nice consequence of existence of entropy for the cellular automata traffic models we are considering here, we can follow microcanonical equilibrium statistical mechanics to find the equilibrium states of these models as the states that maximize the entropy for given density and energy. These constraints seem suitable because cellular automata traffic models involve rules with parameters (such as randomization p in the NS-model) that control the kinetic energy of the particles, and drive the system towards macroscopic steady-states with velocity distribution densities (and kinetic energies) well defined.

By employing the method of undetermined Lagrange multipliers, the velocity distribution functions (or partial densities) n_v which define the equilibrium states were obtained

$$n_v = \left(\frac{\lambda}{\lambda + n}\right)^v e^{-\alpha - \beta \varepsilon_v} \tag{9}$$

where α and β are Lagrange multipliers whose physical meaning is discussed in Section 5. For $v_{max} = 1$, Eqn. (9) leads to the following expressions for the partial densities n_0 and n_1 as functions of n and a energy related parameter γ,

$$n_0 = n - n_1 \tag{10}$$

$$n_1 = \frac{1}{2}\left[1 - \sqrt{1 - 4n(1-n)\left(\frac{1}{\gamma + 1}\right)}\right] \tag{11}$$

Parameter γ was defined in terms of the Lagrange multiplier β which, as we will see later, is the conjugated variable of the global energy density (kinetic energy per site) of the system:

$$\gamma \equiv e^{\beta/2} \tag{12}$$

Traffic flow, in this case, is given by $q = n_1$. Diagrams of equilibrium partial densities n_0 and n_1 are shown in Fig. 11 as functions of n for those models in GC-1DTCA with $v_{max} = 1$, and values $\gamma = 0$, *1/3*, *1*, and *3*. As can be seen, the equilibrium states we have found for $\gamma = 0$ are equivalent to the steady states of the WR184 model (and also to the steady states of the NS-model with randomization $p = 0$, and of the FI-model with stochastic delay $p = 0$). In fact, as we will see later, the equilibrium states given by the equations (10) and (11) are completely equivalent to the steady states of the NS-model with $v_{max} = 1$.

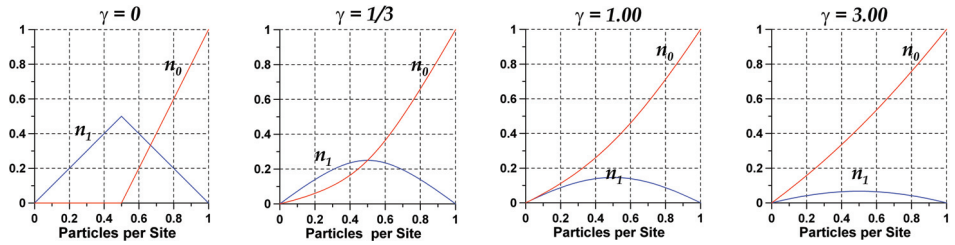


Fig. 11. Equilibrium partial densities n_0 and n_1 as functions of n for models with $v_{max} = 1$. Traffic flow q is the same as n_1. The effect of the parameter γ is appreciated clearly. The equilibrium states of the models in GC-1DTCA with $v_{max} = 1$ and $\gamma = 0$ are equivalent to the steady-states of models WR184, NS (with randomization $p = 0$), and FI (with stochastic delay $p = 0$).

For $v_{max} = 2$, Eqn. (9) lead to the following equations for the velocity distribution densities:

$$n_0 = n - n_1 - n_2 \tag{13}$$

$$n_1 = \frac{(n - n_1 - n_2)(1 - n - n_1 - 2n_2)}{\gamma(1 - n_1 - 2n_2)} \tag{14}$$

$$n_2 = \frac{n_1(1 - n - n_1 - 2n_2)}{\gamma^3(1 - n_1 - 2n_2)} \tag{15}$$

Here, given n and γ, the solution of Eqns. (14) and (15) give n_1 and n_2, and then Eqn. (13) gives n_0. This system of non-linear and coupled equations can be solved with standard numerical methods (See, for example, Press et al., 1992). Here, γ is the same as in Eqn. (12), but now traffic flow, in according to Eqn. (2), is given by $q = n_1 + 2n_2$. Numerical solutions for the equilibrium partial densities n_0, n_1 and n_2 as functions of n are shown in Fig. 12.

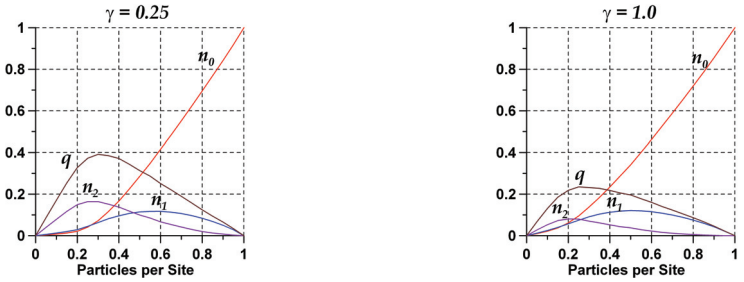


Fig. 12. Equilibrium partial densities n_0, n_1 and n_2 as functions of n for models with $v_{max} = 2$. Traffic flow is $q = n_1 + 2n_2$. The effect of the parameter γ is appreciated clearly.

For $v_{max} > 1$, a singular behaviour of entropy at $n = n_c(v_{max})$ (see Eqn. (1)) becomes evident in the high-energy region ($\beta < 0$). This is shown in Fig. 13 for $v_{max} = 2$. For low energies ($\beta > 0$), the state of the system corresponds to arrangements of particles with the three possible speeds (n_0, n_1, $n_2 > 0$) for all densities $0 < n < 1$. For high energies ($\beta < 0$), however, the number of particles with speed $v = 1$ behaves as a decreasing function of energy and dies out in the high-energy limit ($\beta \to -\infty$). Just at this point, the entropy of the system comes out

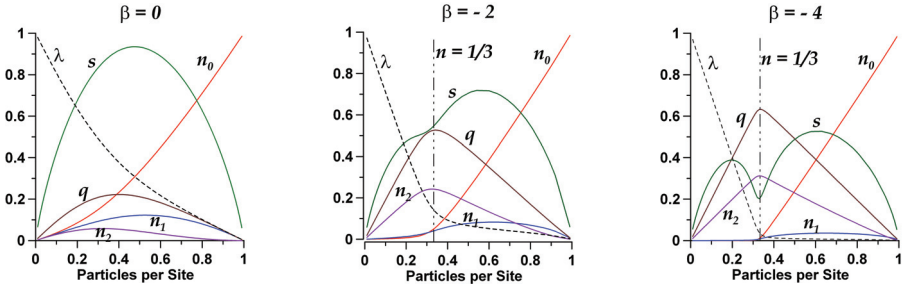


Fig. 13. Equilibrium state entropy, flow and partial densities n_0, n_1 and n_2 as functions of n for models with $v_{max} = 2$. The behaviour of entropy, as energy increases towards its maximum $\varepsilon_{max}(1/3) = 2/3$, shows a flow-regime transition at density $n = n_c(v_{max}) = 1/3$, which becomes sharper for higher energies.

clearly divided in two well differentiated parts: one for densities $n < 1/3$, and the other one for densities $n > 1/3$. The first part corresponds to free-flow states in the system, i.e., states with all particles moving with the speed $v_{max} = 2$, and a number of empty sites $\lambda > 0$. The second part corresponds to congested-flow states in which the system contains only particles with speed v_{max} and particles at rest, and no empty sites ($\lambda = 0$). This behaviour of entropy suggests a flow-regime transition in the system at density $n = n_c(v_{max}) = 1/3$, which becomes sharper when energy increases towards the maximum energy ($\varepsilon_{max}(1/3) = 2/3$).

4. Comparison with the steady states of the NS and FI traffic models

By comparing the simulation results shown in Figs. 4, 7 and 9, we can see that the NS and FI traffic models with $v_{max} = 1$ are identical with each other, and to the WR184 model, in the deterministic setting (i.e. when both the randomization in NS and the stochastic delay in FI are set equal to zero). As it can be observed in the first diagram of Fig. 11, the same behaviour is described by the equilibrium states for $\gamma = 0$, case for which our equations (10) and (11) are reduced to

$$n_0 = n - n_1$$

$$n_1 = \frac{1}{2}\left[1 - \sqrt{1 - 4n(1-n)}\ \right]$$

The NS and FI models, however, behave quite different each other when their respective probability parameters, randomization and stochastic delay, are larger than zero. This is, of course, what one can observe through the comparison of Fig. 7 against the diagrams shown in Fig. 9 (first row). The equilibrium states in this case ($v_{max} = 1$, $p \geq 0$), result expressed by

$$n_0 = n - n_1 \qquad n_1 = \frac{1}{2}\left[1 - \sqrt{1 - 4n(1-n)(1-p)}\ \right] \tag{16}$$

This result is the exact solution for the NS-model with $v_{max} = 1$ (Eqn. (5.11) in Schreckenberg et al, 1995). It is obtained from Eqns. (10) and (11) once probability parameter p is defined as

$$p \equiv \frac{\gamma}{\gamma + 1} = \frac{e^{\beta/2}}{1 + e^{\beta/2}} \tag{17}$$

This is consistent, of course, with the meaning of the randomization probability p in the NS model because, as we will see later, the Lagrange multiplier β is associated with the energy of the system in such a way that $\beta \to -\infty$ ($\beta \to +\infty$) corresponds to the high-energy (low-energy) limit. In fact, we see that the high ($\beta \to -\infty$) and low ($\beta \to +\infty$) energy limits correspond to the randomization limits $p \to 0$ (no braking is allowed at all, and the particles are driven to move with the possible highest speeds) and $p \to 1$ (each particle is obligated to reduce its speed by one each timestep), respectively. In consequence, we are compelled to conclude that the equilibrium states given by the equations (10) and (11) are completely equivalent to the steady states of the NS-model with $v_{max} = 1$.

In Fig. 14, in terms of the partial densities n_v and the traffic flow q, it is shown a comparison of the equilibrium states (solid lines) against computer simulations of the steady states of the NS-model (dashed lines), considering $v_{max} = 2$ and randomizations $p = 0.2$ and 0.5. Here, for each density n we calculated the kinetic energy ε from the partial densities n_v of the

simulated steady state. Then, for each couple (n, ε) we solved numerically the equations (14), (15) and (16). As it can be seen, a reasonable qualitative and quantitative agreement between the equilibrium states and the steady states of NS-model is found. However, growing differences are observed as p decreases below $p = 0.5$, particularly for densities $n > 1/3$.

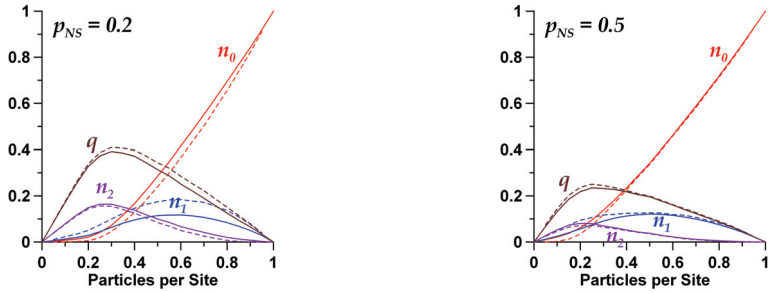


Fig. 14. Comparison between the equilibrium theory results (solid lines) and computer simulation results (dashed lines) carried out with the NS-model with $v_{max} = 2$, for randomizations $p = 0.2$ (left) and 0.5 (right). Although a reasonable agreement is found, important differences are observed.

The observed differences are due to the rules implemented in the NS model for updating the speeds of the particles, which give a non-equilibrium character to the NS-model. In Fig. 15 we have shown diagrams of the deviations of the values of flow q_{NS} and entropy per site s_{NS} of the NS-model steady states with respect to the corresponding values q_{ES} and s_{ES} of the equilibrium states, for $p = 0.2$ and $p = 0.5$. In both cases, the NS steady-state flow deviations are positive, i.e., q_{NS} is larger than q_{ES} for any density n. For the entropy per site, however, also for any density, the values s_{NS} we calculated with the partial densities of the steady states are lesser than the values s_{ES} we calculated from the equilibrium ones, i.e. the

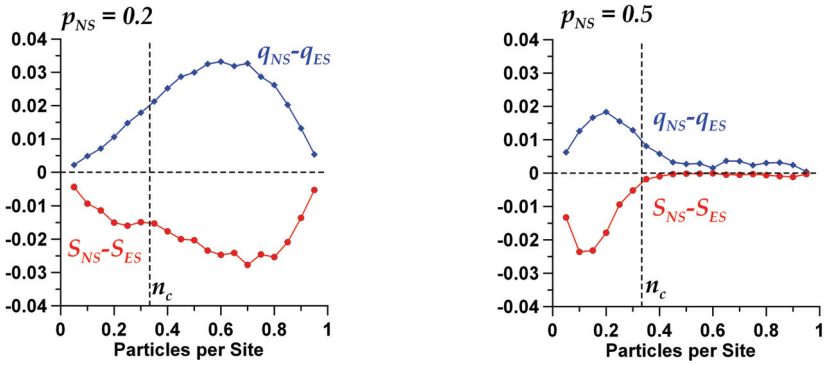


Fig. 15. Deviations of the NS steady state values of traffic flow q (top) and entropy per site s (bottom) with respect the corresponding equilibrium theory values, for $p = 0.2$ (left) and 0.5 (right). In both cases, the NS steady-state flow is larger than that one of equilibrium. For the entropy, however, for any density n, the values calculated with the partial densities of the steady states are smaller than those calculated with the equilibrium ones. This result (we think) is just an expression of the non-equilibrium behaviour of the NS traffic model.

deviations s_{NS} - s_{ES} are negative. This result is exactly what we expected because of the non-equilibrium behaviour of the NS cellular automata traffic model. On another hand, as we see in these figures, for $p = 0.2$, the absolute deviations $|s_{NS} - s_{ES}|$ for density values $n > n_c$ (congested flow regime) are larger than for $n < n_c$ (free-flow regime); while for $p = 0.5$, on the contrary, the absolute deviations for $n < n_c$ are larger than for $n > n_c$. Furthermore, for $p = 0.5$ the system behaviour in the NS-model is very close to equilibrium when $n > n_c$. This is due, of course, to the braking effect of the randomization parameter, which forces a better spreading of the particles among their possible speed values.

5. Equilibrium thermodynamic properties of 1D traffic cellular automata

In order to get some insight about the physical meaning of the Lagrange multipliers α and β, we note that, using Eqn. (9), the entropy can be written as

$$s = (\alpha + \ln \lambda) n + \beta \varepsilon + \ln\left(\frac{\lambda + n}{\lambda}\right) \tag{18}$$

Now, a formal comparison of this equation with the well-known Euler equation of thermodynamics for a gas of particles,

$$s = -\frac{\mu n}{T} + \frac{\varepsilon}{T} + \frac{P}{T} \tag{19}$$

where s is the entropy per unit volume, n is the density of the number of particles, T is the temperature, μ is the chemical potential, ε is the internal energy per unit volume, and P is the pressure, suggests the following thermodynamics interpretation

$$\alpha = -\left(\frac{\mu}{T} + \ln \lambda\right) \tag{20}$$

$$\beta = \frac{1}{T} \tag{21}$$

$$\frac{P}{T} = \ln\left(\frac{\lambda + n}{\lambda}\right) \tag{22}$$

Strictly speaking, Eqns. (20) and (21) just define the new parameters T and μ in terms of the Lagrange multipliers α and β, and equation (22) defines P. However, the use of these properties, which we will call traffic temperature (T), traffic chemical potential (μ), and traffic pressure (P), could open an innovative framework for the physical analysis and interpretation of traffic flow phenomena.

Traffic temperature ($T = 1/\beta$) may assume positive and negative values, with a discontinuity at the kinetic energy ε_c where entropy reaches its maximum (Fig. 16), and splits the energy spectrum in a low ($T > 0$) and a high ($T < 0$) energy regions. This feature of temperature (also inherited by the traffic pressure and chemical potential) is a typical consequence of the upper bound imposed on the kinetic energy of the lattice gas particles (Salcido & Rechtman, 1991; Bagnoli & Rechtman, 2009), and it points out the existence of a population inversion of the particles between the low and high kinetic energies. In fact, as it is suggested by Fig. 16,

as $n \to n_c(v_{max}) = 1/3$, entropy is zero when all the particles are at rest ($n_0 = n$); reaches its maximum value when the numbers of particles at rest and particles in motion are equal: $n_0 = n_1 + n_2$; and becomes null again when all particles are moving with speed v_{max} ($n_2 = n$).

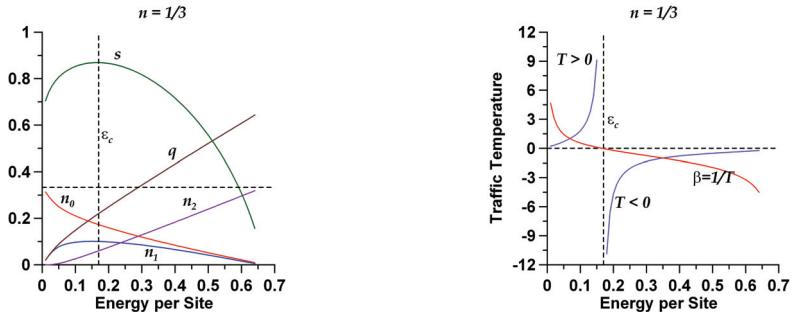


Fig. 16. (Left): The entropy per site (s), flow (q) and partial densities n_v are shown as functions of energy per site (ε) for the critical density $n = n_c(v_{max}) = 1/3$. Observe the typical behaviour of entropy for systems with an upper bound in energy. (Right): Temperature, defined as the slope of entropy as function of energy, has positive values for energies $\varepsilon < \varepsilon_c$, and negative values for $\varepsilon > \varepsilon_c$, being ε_c the energy at which entropy reaches its maximum.

It is interesting the behaviour of traffic pressure in the limits $n \to 0$, $n \to 1$, and $n \to n_c(v_{max})$. For the first two limits, respectively, the Eqn. (22) gives

$$\frac{P}{T} \approx n \qquad \frac{P}{T} \approx \ln\left(\frac{1}{1-n}\right) \tag{23}$$

The result of the first limit ($n \to 0$) resembles the well-known equation of state of an ideal gas. In the second one ($n \to 1$), depending on the s of temperature, $P \to \pm\infty$; this result resembles the behaviour of pressure-density relation in a condensed phase. For high energies ($\beta \to -\infty$), the same as with entropy, traffic pressure and chemical potential have a peculiar behaviour in the limit $n \to n_c(v_{max})$. This is shown in Fig. 17 for speed limit $v_{max} = 2$. In both cases, this behaviour is a result of the singularity these properties have at $\beta = 0$. In the high-energy limit, the density of empty sites dies out ($\lambda \to 0$) when density n increases

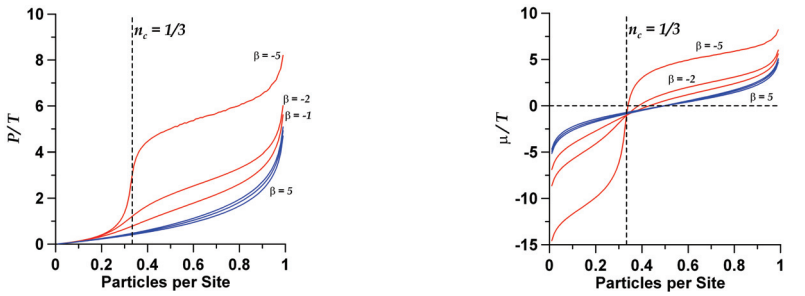


Fig. 17. Properties P/T and μ/T as functions of density n for several values of β. These plots suggest a critical behaviour of traffic flow near $n = n_c(v_{max}) = 1/3$ in the limit of high energy ($\beta \to -\infty$) of the system with $v_{max} = 2$.

towards $n_c(v_{max})$, and so P/T and μ/T diverge to infinity as $ln(1/\lambda)$. Because λ remains null for densities larger than $n_c(v_{max})$ (see Fig. 13c), P/T and μ/T will remain undefined there.
Other thermodynamic properties, such as the specific heat C_v, isothermal compressibility κ_T, and isobaric expansivity α_P, can be calculated easily from the velocity distribution densities. The expressions we have obtained for these thermodynamic properties are shown in the equations (24), (25) and (26). The behaviour of these properties is shown in Fig. 18.

$$C_v \equiv -\beta^2 \left(\frac{\partial \varepsilon}{\partial \beta} \right)_v = \beta^2 \sum_{v=0}^{v_{\max}} n_v \varepsilon_v^2 - \frac{(\beta e)^2}{n} \tag{24}$$

$$\kappa_T(P,\beta) \equiv -\frac{1}{v}\left(\frac{\partial v}{\partial P} \right)_\beta = -\frac{n\beta e^{\beta P}}{(e^{\beta P}-1)(e^{\beta P}-1-n)} \tag{25}$$

$$\alpha_P(P,\beta) \equiv -\beta^2 \frac{1}{v}\left(\frac{\partial v}{\partial \beta} \right)_P = \beta P \kappa_T(P,\beta) \tag{26}$$

with

$$v \equiv 1-\lambda = 1-\frac{n}{e^{\beta P}-1} \tag{27}$$

In this figure, we can observe the effect on the thermodynamic properties due to a sharp transition between the free- and congested-flow regimes. In particular, it is observed that compressibility and expansivity go to zero as n is increased towards $1/3$. This means that for $n > 1/3$, in the high-energy limit, no empty sites will be available in the lattice in order to set in motion the particles at rest (i.e., the system of particles cannot be expanded), and any particle will be found at rest or moving with the speed v_{max}. This behaviour of particles in the high-energy limit is observed also in the Fig. 13.

6. Conclusions and future work

The cellular automata traffic models of Nagel-Schreckenberg (NS) and Fukui-Ishibashi (FI) include velocity updating-rules which define a dynamics that do not obey a detailed balance. These rules continuously drive the system to states out of equilibrium. This is the reason why these models and their variants cannot be studied within the framework of equilibrium statistical mechanics. Nevertheless, as we have shown here, thermodynamic entropy exists for the 1DTCA models with no velocity anticipation, which we have found through an isomorphic system where a lattice gas particle which moves with speed v is modelled as a brick, $(v+1)$-sites length, that is inserted in a 1D lattice with no overlapping. This allowed us to study the equilibrium (or maximum entropy) states of these systems and their thermodynamic properties. As it could be expected, the maximum entropy states do not agree in general with the steady states of the NS model, particularly for high energies (i.e. small values of the randomization parameter p); however, in the low energies domain the equilibrium states resemble very strongly the steady states of the NS model, and the fundamental diagrams are reproduced quit well. For $v_{max} = 2$, the behaviour of entropy, as a function of the density of particles, allowed a clear identification of different flow-regimes in the 1DTCA models, displaying a sharp transition between the free- and congested-flow regimes in the high-energy limit. The presence of this transition was observed also in other

properties of the system, such as the specific heat, the isothermal compressibility, and the isobaric expansivity, which were calculated using the velocity distribution densities of the equilibrium states. Therefore, we presume that the knowledge of thermodynamic properties within the context of modelling traffic flow by means of cellular automata is quite relevant for improving and speeding up the computer simulation of traffic flow, but also may help us to improve the physical understanding of traffic flow phenomena. So, we hope this work may contribute in advancing some modest steps in the establishment of the traffic cellular automata theory.

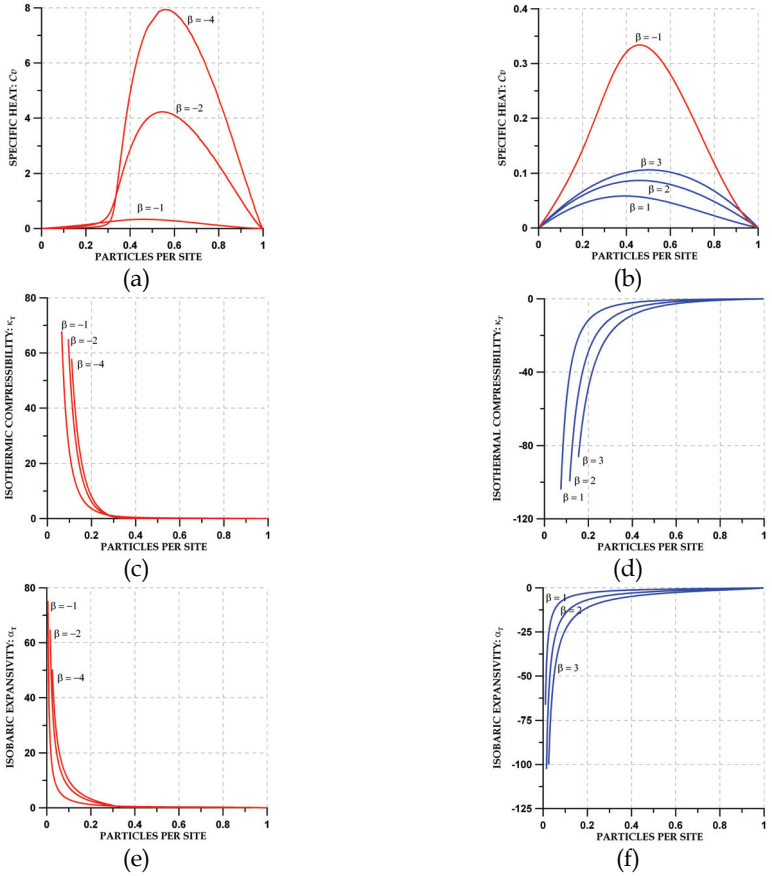


Fig. 18. Behaviour diagrams of the thermodynamic properties of the 1DTCA models with $v_{max} = 2$, for $\beta < 0$ (left column) and $\beta > 0$ (right column). First row: Specific heat C_v. Second row: Isothermal compressibility κ_T. Third row: Isobaric expansivity α_P. All the properties were plotted as functions of the density of particles n. For $\beta < 0$ these diagrams show the occurrence a flow-regime transition at density $n = n_c(v_{max}) = 1/3$.

In the near future, we would like to be able of extending this equilibrium theory to the cellular automata models for multi-lane traffic flow and 2D traffic networks, which we hope would be setting us in the way towards the application of these models within the context of the urban air pollution problems.

7. Acknowledgements

We acknowledge the disinterested help and invaluable contributions we received from A. Fierros Palacios (Director, IIE-DEA, Mexico), R. Merino (TECALCO, Mexico), and A. T. Celada Murillo (IIE-DEA, Mexico) through their encouragements, comments, and enlightening discussions during the preparation of this work.

8. References

Bagnoli, F. (2001). Cellular Automata, In: *Dynamical Modeling in Biotechnology*, F. Bagnoli & S. Ruffo (Eds.), 3-46, World Scientific, ISBN: 981-02-3604-2, Singapore.

Bagnoli, F. & Rechtman, R. (2009). Thermodynamic entropy and chaos in a discrete hydrodynamical system, *Physical Review E,* Vol. 79, 041115 (April, 2009) 041115-1–041115-6, ISSN: 1539-3755.

Batchelor, G. K. (1967). *An Introduction to Fluid Dynamics*. Cambridge University Press, ISBN-13: 978-0521663960, ISBN-10: 0521663962, Cambridge.

Biham, O.; Middleton, A. A. & Levine, D. (1992). Self-organization and a dynamical transition in traffic-flow models, *Physical Review A*, Vol. 46, Issue 10, (November 1992), R6124–R6127, ISSN: 1050-2947.

Boghosian, B. M. (1999). Lattice Gases and Cellular Automata, *Future Generation Computer Systems*, Vol. 16, Issues 2-3, (December 1999), 171-185. ISSN: 0167-739X.

Boccara, N. (2001). On the existence of a variational principle for deterministic cellular automaton models of highway traffic flow, *International Journal of Modern Physics C (IJMPC)*, Vol. 12, Issue 2, (February 2001), 143-158, ISSN: 0129-1831.

Blank, M. (2005). Hysteresis phenomenon in deterministic traffic flows, *Journal of Statistical Physics,* Vol. 120, Issues 3-4, (2005), 627-658, ISSN: 0022-4715.

Blank, M. (2008). Travelling with/against the Flow. Deterministic Diffusive Driven Systems, *Journal of Statistical Physics,* Vol., 133, Issue 4, (2008), 773-796, ISSN: 0022-4715.

Brilon, W. & Wu, N. (1999). Evaluation of cellular automata for traffic flow simulation on freeway and urban streets, In: *Traffic andMobility: Simulation-Economics-Environment*, W. Brilon, F. Huber, M. Schreckenberg, and H. Wallentowitz (Eds.), 163–180, Springer, ISBN-10: 3540662952, ISBN-13: 978-3540662952, Berlin.

Chen, S.; Lee, M.; Zhao, K. H. & Doolen, G. D. (1989). A Lattice Gas Model with Temperature, *Physica D: Nonlinear Phenomena*, Vol. 37, Issues 1-3, (July 1989), 42-59, ISSN: 0167-2789.

Chopard, B. & Droz, M. (1998). *Cellular Automata Modeling of Physical Systems*, Cambridge University Press, ISBN: 0 521 46168 5, Cambridge.

Chowdhury, D.; Santen, L. & Schadschneider, A. (2000). Statistical Physics of Vehicular Traffic and Some Related Systems. *Physics Reports*, Vol. 329, Issues 4-6, (May 2000) 199-329, ISSN: 0370-1573.

Cremer, M. & Ludwig, J. (1986). A fast simulation model for traffic flow on the basis of boolean operations, *Mathematics and Computers in Simulation*, Vol. 28, Issue 4 (August 1986), 297-303, ISSN: 0378-4754.

d´Humieres, D.; Lallemand, P. & Frish, U. (1986). Lattice Gas Models for 3D Hydrodynamics, *Europhysics Letters*, Vol. 2, Issue 4, (August 1986), 291-297, ISSN: 0295-5075.

Domb, C. & Lebowitz, J.L. (Eds.) (2001). Phase Transitions and Critical Phenomena, Vol. 19 (Academic Press, USA, 2001).

Doolen, G.D.; Frisch, U.; Hasslacher, B.; Orszag, S. & Wolfram, S. (Eds). (1990). *Lattice Gas Methods for Partial Differential Equations*, Addison Wesley, ISBN: 020113232X, USA.

Esser, J. & Schreckenberg, M. (1997). Microscopic Simulation of Urban Traffic based on Cellular Automata, *International Journal of Modern Physics C (IJMPC)*, Vol. 8, Issue 5, (1997), 1025-1036, ISSN: 0129-1831.

Evans, M. R.; Rajewsky, N. & Speer, E. R. (1999). Exact Solution of a Cellular Automaton for Traffic, *Journal of Statistical Physics,* Vol. 95, Issues 1-2 (April 1999), 45-96, ISSN: 0022-4715.

Frish, U.; Hasslacher, B. & Pomeau, Y. (1986). Lattice-Gas Automata for the Navier-Stokes Equation, *Physical Review Letters*, Vol. 56, Issue 14, (April 1986), 1505-1508, ISSN: 1079-7114.

Fukui, M. & Ishibashi, Y. (1993). Evolution of Traffic Jam in Traffic Flow Model, *Journal of the Physical Society of Japan*, Vol. 62, (1993), 3841-3844, ISSN: 0031-9015.

Fukui, M. & Ishibashi, Y. (1996a). Traffic Flow in 1D Cellular Automaton Model Including Cars Moving with High Speed, *Journal of the Physical Society of Japan*, Vol. 65, (1996), 1868-1870, ISSN: 0031-9015.

Fukui, M. & Ishibashi, Y. (1996b). Effect of reduced randomness on jam in a two-dimensional traffic model, *Journal of the Physical Society of Japan*, Vol. 65, (1996), 1871-1873, ISSN: 0031-9015.

Fukui, M.; Oikawa, H. & Ishibashi, Y. (1996). Flow of cars crossing with unequal velocities in a two-dimensional cellular automaton model. *Journal of the Physical Society of Japan*, Vol. 65, (1996), 2514-2517, ISSN: 0031-9015.

Fuks, H. (1999). Exact results for deterministic cellular automata traffic models, *Physical Review E*, Vol. 60, Issue 1, (July 1999), 197-202, ISSN: 1539-3755.

Gazis, D. C. (1967). Mathematical Theory of Automobile Traffic: Improved understanding and control of traffic flow has become a fast-growing area of scientific research, *Science*, Vol. 157, No. 3786, (July 1967) 273 – 281, ISSN: 0036-8075.

Hardy, J.; de Pazzis, O. & Pomeau, Y. (1976). Molecular Dynamics of a Classical Lattice Gas: Transport Properties and Time Correlation Functions, *Physical Review A*, Vol. 13, Issue 5, (May 1976), 1949-1961, ISSN: 1050-2947.

Hardy, J.; Pomeau, Y. & de Pazzis, O. (1973). Time Evolution of a Two-Dimensional Model System. I. Invariant States and Time Correlation Functions, *Journal of Mathematical Physics*, Vol. 14, Issue 12, (December 1973), 1746-1759, ISSN: 0022-2488.

Hasslacher, B. (1987). Discrete Fluids, *Los Alamos Science*, Vol. 15, Special Issue, (1987), 175-217. Available at: www.fas.org/sgp/othergov/doe/lanl/pubs/00285743.pdf

Helbing, D. (1996). Gas-kinetic derivation of Navier-Stokes-like traffic equations, *Physical Review E,* Vol. 53, Issue 3, (March 1996) 2366–2381, ISSN: 1539-3755.

Helbing, D. (1998). Structure and instability of high-density equations for traffic flow, *Physical Review E,* Vol. 57, Issue 5, (May 1998) 6176-6179, ISSN: 1539-3755.

Helbing, D. (2001). Traffic and related self-driven many-particle systems, *Reviews of Modern Physics,* Vol. 73, Issue 4, (October 2001) 1067–1141, ISSN: 0034-6861.

Helbing, D. & Treiber, M. (1998). Gas-Kinetic-Based Traffic Model Explaining Observed Hysteretic Phase Transition, *Physical Review Letters*, Vol. 81, Issue 14, (October 1998) 3042-3045, ISSN: 0031-9007.

Herman, R. & Gardels, K. (1963). Vehicular Traffic Flow, *Scientific American*, Vol. 209, Issue 6, (1963), 35-43.

Kadanoff, L. P. & Swift, J. (1968). Transport Coefficients near the Critical Point: A Master-Equation Approach, *Physical Review*, Vol. 165, Issue 1, (January 1968), 310-322, ISSN: 0031-899X.

Krauß, S.; Nagel, K. & Wagner, P. (1999). The mechanism of flow breakdown in traffic flow models, in: Proceedings of the International Symposium on Traffic and Transportation Theory (ISTTT99), Jerusalem, 1999.

Kühne, R. & Michalopoulos, P. (1998). Continuum Flow Models, In: *Traffic Flow Theory. A State of the Art Report*, N. Gartner, C.J. Messner & A.J. Rathi (Eds.), Transportation Research Board (TRB) Special Report 165, 2nd ed.

Lee, K.; Hui, P.; Mao, D.; Wang, B. H. & Wu, Q. S. (2002). Fukui–Ishibashi traffic flow models with anticipation of movement of the car ahead, *Journal of the Physical Society of Japan*, Vol. 71, No. 7, (February 2002), 1651–1654, ISSN: 0031-9015.

Maerivoet, S. & De Moor, B. (2005). Cellular automata models of road traffic, *Physics Reports* Vol. 419, Issue 1, (November 2005), 1-64, ISSN; 0370-1573.

MM5. (2003). MM5 Community Model. Visit: http://www.mmm.ucar.edu/mm5/

Nagatani, T. (1997a). Kinetic segregation in a multilane highway traffic flow, *Physica A* Vol. 237, Issues 1-2, (March 1997), 67-74, ISSN: 0378-4371.

Nagatani, T. (1997b). Gas Kinetics of Traffic Jam, *Journal of the Physical Society of Japan*, Vol. 66, (1997), 1219-1224, ISSN: 0031-9015.

Nagatani, T. (2002). The physics of traffic jams, *Reports on Progress in Physics*, Vol. 65, Issue 9, (September 2002), 1331-1386, ISSN: 0034-4885.

Nagel, K. (1996). Particle hopping models and traffic flow theory, *Physical Review E*, Vol. 53, Issue 5, (May 1996), 4655-4672, ISSN: 1539-3755.

Nagel, K. & Barrett, C. L. (1997). Using Microsimulation Feedback For Trip Adaptation For Realistic Traffic In Dallas, *International Journal of Modern Physics C*, Vol. 8, Issue 3, (June 1997), 505-525, ISSN: 0129-1831.

Nagel, K. & Nelson, P. (2005). A critical comparison of the kinematic-wave model with observational data, In: *Transportation and Traffic Theory. Flow, Dynamics and Human Interaction*, H.S. Mahmassani (Ed.), 145-163, Elsevier, ISBN: 0-08-044680-9, USA.

Nagel, K. & Paczuski, M. (1995). Emergent traffic jams, *Physical Review E*, Vol. 51, Issue 4, (April 1995), 2909-2918, ISSN: 1539-3755.

Nagel, K. & Schreckenberg, M. (1992). A cellular automaton model for freeway traffic. *Journal de Physique I*, France, Vol. 2, No. 12, (December 1992), 2221-2229.

Nagel, K.; Wagner, P. & Woesler, R. (2003). Still Flowing: Approaches to Traffic Flow and Traffic Jam Modeling, *Operations Research*, Vol. 51, No. 5, (September-October 2003), 681-710, ISSN: 0030-364X.

Nagel, K.; Wolf, D. E.; Wagner, P. & Simon, P. (1998). Two-lane traffic rules for cellular automata: A systematic approach, *Physical Review E,* Vol. 58, Issue 2, (August 1998) 1425–1437, ISSN: 1539-3755.

Press, W.H; Teukolsky, S.A.; Vetterling, W.T. & Flannery, B.P. (1992). *Numerical Recipes in C: The Art of Scientific Computing*, Chapter 9, Section 9.7, Cambridge University Press, ISBN: 0-521-43108-5, Cambridge.

Prigogine, I. & Herman, R. (1971). *Kinetic Theory of Vehicular Traffic.* American Elsevier, ISBN: 0-444-00082-8, New York.

Rechtman, R. & Salcido, A. (1996). Lattice Gas Self Diffusion in Random Porous Media, *Fields Institute Communications*, Vol. 6, (1996), 217-225, ISSN: 1069-5265.

Rechtman, R.; Salcido, A. & Bagnoli, F. (1990). Thermomechanical Effects in a Nine-Velocities Two-Dimensional Lattice Gas Automaton, In: *Lectures on Thermodynamics and Statistical Mechanics*, M. López de Haro and C. Varea (Eds.), 182-200, World Scientific., ISBN 981-02-0243-1, Singapore.

Rechtman, R.; Salcido, A. & Bagnoli, F. (1992). Some Near-Equilibrium Properties of a Nine-Velocities Lattice Gas Automaton for Two-Dimensional Hydrodynamics, In: *Complex Dynamics*, R. Livi, J-P. Nadal and N. Packard (Eds.), 133-139, Nova Science Publishers Inc., ISBN: 1560720182, New York.

Rickert, M. & Nagel, K. (1997). Experiences with a Simplified Microsimulation for the Dallas/Fort-Worth Area, *International Journal of Modern Physics C*, Vol. 8, Issue 3, (June 1997), 483-503, ISSN: 0129-1831.

Rothery, R. W. (1998). Car Following Models. In: *Traffic Flow Theory. A State of the Art Report*, N. Gartner, C.J. Messner & A.J. Rathi (Eds.), Transportation Research Board (TRB) Special Report 165, 2nd ed.

Rothman, D. & Zaleski, S. (1997). Lattice-Gas Cellular Automata, Simple Models of Complex Hydrodynamics, Cambridge University Press, ISBN: 0-521-55-201-X, Cambridge.

Salcido, A. (1993). Lattice Gas Model for Transport and Dispersion Phenomena of Air Pollutants, In: *Transactions on Ecology and the Environment*, Vol. 1, P. Zannetti, C.A. Brebbia, J.E. Garcia Gardea and G. Ayala Milian (Eds.), 173-181, WIT Press, ISSN: 1743-3541, Southampton.

Salcido, A. (1994). First Evaluations of a Lattice Gas Approach to Air Pollution Modelling. In: *Transactions on Ecology and the Environment*, Vol. 3, J. M. Baldasano, C. A. Brebbia, H. Power and P. Zannetti (Eds.), 141-150, WIT Press, ISSN 1743-3541, Southampton.

Salcido, A. (2007). The Maximum Entropy States of 1D Cellular Automata Traffic Models, *Proceedings of the 18th IASTED International Conference on Modelling and Simulation*, 160-165. Montreal, Canada. 2007. ACTA Press Anaheim, CA, USA. ISBN: 978-0-88986-664-5.

Salcido, A. & Celada-Murillo, A. T. (2010). A Lattice Gas Approach to the Mexico City Wind Field Estimation Problem. In: *Modelling, Simulation and Optimization*, G. Romero Rey and L. Martinez Munela (Eds.), 385-416, In-The, ISBN: 978-953-307-048-3, Croatia.

Salcido, A.; Celada-Murillo, A. T. & Castro, T. (2008). Lattice Gas Simulation of Wind Fields in the Mexico City Metropolitan Area, *Proceedings of the 19th IASTED International Conference on Modelling and Simulation (MS 2008)*, 95-100, ISBN: 9780889867413, Quebec, Canada, May 2008, Acta Press, Anaheim, Calgary, Zurich.

Salcido, A.; Merino, R. & Saldaña, R. (1993). Lattice Gas Model for Wind Fields over Complex Terrains. *Proceedings of the International Symposium on Heat and Mass Transfer in Energy Systems and Environmental Effects*, 526-531, Cancun, Mexico, August 1993, International Centre for Heat and Mass Transfer, Cancun.

Salcido, A. & Rechtman, R. (1991). Equilibrium properties of a cellular automaton for thermofluid dynamics. In: *Nonlinear Phenomena in Fluids, Solids and Other Complex Systems*, P. Cordero and B. Nachtergaele (Eds), 217-229, Elsevier, ISBN: 0444887911, ISBN-13: 9780444887917, Amsterdam.

Salcido, A. & Rechtman, R. (1993). Lattice Gas Simulations of Flows Through Two-Dimensional Porous Media, *Proceedings of the International Symposium on Heat and Mass Transfer in Energy Systems and Environmental Effects*, 222-226, Cancun, Mexico, August 1993, International Centre for Heat and Mass Transfer, Cancun.

Sasvári, M. & Kertész, J. (1997). Cellular automata models of single-lane traffic, *Physical Review E*, Vol. 56, Issue 4, (October 1997), 4104–4110, ISSN: 1539-3755.

Schadschneider, A. (1999). The Nagel-Schreckenberg model revisited, *The European Physical Journal B*, Vol. 10, Number 3, (August_I 1999), 573-582, ISSN: 1434-6028.

Schadschneider, A. & Schreckenberg, M. (1993). Cellular automaton models and traffic flow, *Journal of Physics A: Mathematical and General*, Vol. 26, Issue 15, (August 1993), L679-L684, ISSN: 0305-4470.

Schadschneider, A. & Schreckenberg, M. (1997). Car-oriented mean-field theory for traffic flow models, *Journal of Physics A: Mathematical and General*, Vol. 30, Issue 4, (February 1997), L69-L75, ISSN: 0305-4470.

Schreckenberg, M.; Schadschneider, A.; Nagel, K. & Ito, N. (1995). Discrete stochastic models for traffic flow, *Physical Review E*, Vol. 51, Issue 4, (April 1995), 2939-2949, ISSN: 1539-3755.

Schütt, H. (1991). Entwicklung undErprob ung eines sehr schnellen, bitorientierten Verkehrssimulationssystems für Straßennetze, Technical Report No. 6, Schriftenreihe der AG Automatisierungstechnik, T.U. Hamburg, Hamburg, 1991.

Sciarretta, A. (2006). A lattice gas model with temperature and buoyancy effects to predict the concentration of pollutant gas released by power plants and traffic sources, *Mathematical and Computer Modelling of Dynamical Systems*, Vol. 12, Issue 4, (August 2006), 313-327, ISSN: 1387-3954.

Sciarretta, A. & Cipollone, R. (2001). A lattice gas model for the evaluation of transport and diffusion parameters of stack emissions in air. In: *Air Pollution IX*, G. Latini and C. A. Brebbia (Eds), WIT Press, ISBN: 1853128775, Southampton.

Sciarretta, A. & Cipollone, R. (2002). On the evaluation of pollutant gas dispersion around complex sources by means of a lattice gas model. In: *Air Pollution X*, C. A. Brebbia and J. Martin-Duque (Eds.), 33-42, WIT Press, ISBN: 185312916X, Southampton.

Schütz, G. M. (2001). Exactly Solvable Models for Many-Body Systems Far from Equilibrium, In: *Phase Tansitions and Critical Phenomena*, Vol. 19, C. Domb and J.L. Lebowitz (Eds.), 1-251, Academic Press, ISBN: 0-12-220319-4, San Diego, Ca, USA.

Simon, P. M. & Nagel, K. (1998). A Simplified Cellular Automaton Model for City Traffic, *Physical Review E*, Vol. 58, Issue 2, (August 1998), 1286-1295, ISSN: 1539-3755.

SMA-GDF (2008). Inventario de Emisiones de Contaminantes Criterio de la Zona Metropolitana del Valle de México, 2006. Primera Edición. Secretaría del Medio Ambiente. Gobierno del Distrito Federal. México. Document available from:

http://www.sma.df.gob.mx/sma/links/download/archivos/ie06_criterio_pw23oct08.pdf

Stauffer, D. (2001). Cellular Automata: Applications. In: *Vector and Parallel Processing-VECPAR 2000*, J. Palma, J. Dongarra & V. Hernández (Eds.), *Lecture Notes in Computer Science,* Vol. 1981/2001, (2001), 199-206, ISSN: 0302-9743. Springer Berlin/Heidelberg.

Toffoli, T. (1984). Cellular automata as an alternative to (rather than an approximation of) differential equations in modeling physics, Physica D, 10, 1, (January 1984) 117-127, ISSN: 0167-2789.

von Neumann, J. (1951). The general and logical theory of automata. In: *Cerebral Mechanisms in Behaviour: The Hixon Symposium*, L.A. Jeffress (Ed.), (1951), John Wiley, New York. (Reprinted in *J. von Neumann, Collected Works*, A. H. Taub (Ed.), Vol. 5, (1963), 288–328, Pergamon Press, New York)

von Neumann, J. (1966). *Theory of Self-Reproducing Automata*, Edited and completed by A.W. Burks University of Illinois Press, ISBN: 0252727339 (ISBN13: 9780252727337), Urbana, Illinois.

Vilar, L. C. Q. & de Souza, A. M. C. (1994). Cellular automata models for general traffic conditions on a line, *Physica A: Statistical Mechanics and its Applications,* Vol. 211, Issue 1, (October, 1994), 84-92, ISSN: 0378-4371.

Wagner, P.; Nagel, K. & Wolf, D. E. (1997). Realistic multi-lane traffic rules for cellular automata, *Physica A: Statistical Mechanics and its Applications*, Vol. 234, Issue 3-4, (January 1997), 687-698, ISSN: 0378-4371.

Wang, B. H.; Kwong, Y.R. & Hui, P. M. (1998a). Statistical mechanical approach to Fukui–Ishibashi traffic flow models, *Physical Review E*, Vol. 57, Issue 3, (March, 1998), 2568–2573, ISSN: 1539-3755.

Wang, L.; Wang, B. H. & Hu, B. (2001). A cellular automaton traffic flow model between the Fukui–Ishibashi and Nagel–Schreckenberg models, *Physical Review E*, Vol. 63, Issue 5, (April, 2001), [5 pages]: 056117, ISSN: 1539-3755.

Wang, B. H., Wang, L. & Hui, P.M. (1997). One-Dimensional Fukui-Ishibashi Traffic Flow Model, *Journal of the Physical Society of Japan*, Vol. 66, No. 11, (November 1997) 3683-3684, ISSN: 0031-9015.

Wang, B. H.; Wang, L.; Hui, P. M. & Hu, B. (1998b). Analytical results for the steady state of traffic flow models with stochastic delay, *Physical Review E*, Vol. 58, Issue 3, (September 1998), 2876-2882, ISSN: 1539-3755.

Wolfram, S. (1984). Universality and Complexity in Cellular Automata, *Physica D : Nonlinear Phenomena*, Vol. 10, Issues 1-2, (January, 1984), 1-35, ISSN: 0167-2789.

Wolfram, S. (1986a). Cellular Automaton Fluids I. Basic Theory, *Journal of Statistical Physics,* Vol. 45, Issues 3-4, (November 1986), 471-526, ISSN: 0022-4715.

Wolfram, S. (1986b). *Theory and Applications of Cellular Automata*. World Scientific, Singapore.

Wolfram, S. (1994). *Cellular Automata and Complexity: Collected Papers*. Addison-Wesley, Reading, Massachusetts.

Wolfram, S. (2002). *A New Kind of Science*. Wolfram Media, Inc, 2002.

Zannetti, P. (1990). *Air Pollution Modelling. Theories, Computational Methods and Available Software*, Computational Mechanics Publications, ISBN: 0442308051, Southampton, Boston, New York.

9

Cellular Automata for Traffic Modelling and Simulations in a Situation of Evacuation from Disaster Areas

Kohei Arai, Tri Harsono and Achmad Basuki
Saga University
Japan

1. Introduction

The traffic flow studies using microscopic simulations (micro traffic model) have leap with the occurrence of the advancement of computer technology in the last one and half decade, as shown in (Nagel, K. & Schreckenberg, M., 1992); (Nagel, K., 1996); (Bando, M., et al., 1995).
The evacuation system in the micro traffic simulation model has been studied and reported a couple of years ago. In the early stage, some examples of micro traffic simulation models related with emergency evacuation are investigated by (Sugiman, T. & Misumi, J., 1988); (Stern, E. & Sinuany-Stem, Z., 1989). The modelling system of emergency evacuation in the traffic stated by (Sheffi, Y., et al., 1982); (Hobeika, A.G. & Jamei, B., 1985); (Cova, T.J. & Church, R.L., 1997) has chosen to estimate evacuation time from an affected area using static analysis tools at macroscopic or microscopic levels.
Another research of emergency evacuation at the micro traffic scale was conducted by (Pidd, M., et al., 1996). They developed a prototype of spatial decision support system that can be used for emergency planners to evaluate contingency plans for evacuation from disaster areas. It does not take the interactions between individual vehicles into consideration.
Two basic components of agent-based modeling, i.e. (1) a model of agents and (2) a model of their environment were introduced by (Deadmann, P.J., 1999). An individual agent makes a decision based on the interaction between him and the other agents together with localized knowledge (Teodorovic, D.A., 2003).
How evacuation time can be affected under different evacuation scenarios, such as opening an alternative exit, invoking traffic control, changing the number of vehicles leaving a household was observed based on agent-based simulation techniques (Church, R.L. & Sexton, R., 2002). Neighbourhood evacuation plans in an urbanized wild land interface were described by (Cova, T.J. & Johnson, J.P., 2002) using agent-based simulation model. They were able to assess spatial effects of a proposed second access road on household evacuation time in a very detailed way.
The effectiveness of simultaneous and staged evacuation strategies using agent-based simulation for three different road network structures were presented by (Chen, X. & Zhan, F.B., 2008). They measured the effectiveness based on total time of evacuation from affected areas.

Aforementioned studies described how to evacuate all residents in affected area whereas this study evacuates vehicles in affected road using agent-based modelling. We conduct micro traffic agent-based modelling and simulation for assessment of evacuation time with and without agents from the suffered area of the Sidoarjo hot mudflow situated in the East Java Indonesia called LUSI that occurred on 29th of May, 2006. Even now, the mud volcano remains high flow rates (Rifai, R., 2008).
One of the key elements of evacuation from the mudflow disaster is the road as main traffic connection surrounding disaster area and dike of the hot mudflow is very close with the road (Indahnesia.com, 2007). The vehicles density on the road is high (Mediacenter, 2007).
Our micro traffic agent-based modelling and simulation, other than with/without agent; road traffic (vehicle density) and road networks; driving behaviour such as lane changing, car-following; and unpredictable disturbance due to a difference between disaster speed and vehicle speed are taken into account. Although the proposed simulation is based on the Nagel-Schreckenberg proposed by traffic Cellular Automata (CA) as shown in (Nagel, K. & Schreckenberg, M., 1992) and (Maerivoet, S. & De Moor, B., 2005), lane changing and car-following parameters are specific to the proposed simulation.
The following section describes overview of the traffic models followed by car-following models. Then the introduction of CA model, major topic of the chapter of evacuation simulation will be explained. Agent-based approach of CA traffic model for the case of evacuation is presented. Furthermore, the results of traffic survey are described. The traffic survey is conducted on the specific roadway situated surrounding the hot mudflow disaster area, Sidoarjo, Indonesia. It is found two different types of driver, i.e. usual diver and diligent one. The next section, modified car-following model is investigated by taking the behavior of usual driver and diligent one into account. Finally, concluding remarks and some discussions are described.

2. Overview of the traffic models

One important portion of the micro traffic model is Car-Following (CF). Studies of car-following models have been proposed to describe the interaction between drivers and vehicles. CF became an important evaluation parameter for intelligent transportation system strategies since 1990. CF theories based on the assumption that each driver reacts in some specific way to be stimulated with the vehicles in front. A very attractive microscopic traffic model called the Optimal Velocity Model (OVM) is proposed by (Bando, M., et al., 1995). It was based on the idea that each vehicle has an optimal velocity, which depends on the following distance of the preceding vehicle. Despite its simplicity and its few parameters, the OVM can describe many properties of real traffic flows, such as the instability of traffic flow, the evolution of traffic congestion, and the formation of stop-and-go waves.
OVM used to be calibrated using empirical data provided by (Helbing, D. & Tilch, B., 1998). They stated that when empirical data used in OVM has weaknesses, OVM has a too high acceleration and an unrealistic deceleration. (Helbing, D. & Tilch, B., 1998) have overcome these problems by using a Generalized Force Model (GFM).
(Jiang, R., et al., 2001) stated that GFM cannot describe the delay time δt and the kinematics wave speed at jam density cj properly. They so that proposed the Full Velocity Difference Model (FVDM). FVDM has too high deceleration since empirical deceleration and acceleration are restricted between the following regions, from $-3m/s^2$ to $4m/s^2$.

On the other hand about improving the OVM and considering the Intelligent Transportation System (ITS) application, (Ge, X.H., et al., 2008) proposed a new model taking into account the velocity difference Δv_n and Δv_{n+1}, where $\Delta v_n \equiv v_{n+1} - v_n$. They obtain a more useful model called the Two Velocity Difference model (TVDM). TVDM has shown that unrealistic high deceleration does not appear when they simulate the deceleration progress of two cars.
The CF models (Bando, M., et al., 1995); (Helbing, D. & Tilch, B., 1998); (Jiang, R., et al., 2001) and (Ge, X.H., et al., 2008) are known as time-continuous models. They have in common that they are defined by ordinary differential equations describing the complete dynamics of the vehicle's positions x and velocity v.
Vehicle dynamics system can also be expressed by CA model. The road is divided into sections of a certain length Δx and the time is discretised to steps of Δt. Each road section can either be occupied by a vehicle or empty and the dynamics are given by update rules of the form: $v_n^t = f\left(x_n^{t-1}, v_n^{t-1}, v_{n+1}^{t-1}, \cdots\right)$ and $x_n^t = x_n^{t-1} + v_n^t$, the simulation time t is measured in units of Δt and the vehicle positions x_n is measured in units of Δx.
In this study, we proposed the vehicle dynamics system based on Stochastic Traffic Cellular Automaton (STCA) expressed in (Nagel, K. & Schreckenberg, M., 1992). It has the ability to reproduce a wide range of traffic phenomena. Due to the simplicity of the models, it is numerically very efficient and can be used to simulate large road networks or even faster. A new characteristic of driving behaviour has been proposed, it is about diligent driver.
The proposed evacuation modelling and simulations above (Sugiman, T. & Misumi, J., 1988); (Stern, E. & Sinuany-Stem, Z., 1989); (Sheffi, Y., et al., 1982); (Hobeika, A.G. & Jamei, B., 1985); (Cova, T.J. & Church, R.L., 1997); (Church, R.L. & Sexton, R., 2002); (Cova, T.J. & Johnson, J.P., 2002) and (Chen, X. & Zhan, F.B., 2008) describe how to evacuate all residents from affected areas, the proposed modelling and simulation here is for vehicle evacuation from affected road based on an agent-based modelling. Namely, we conduct micro traffic agent-based modelling and simulation for assessment of evacuation time from the suffered area. In the micro traffic agent-based modelling and simulation, road traffic (probability of vehicle density), driving behaviour such as probability of lane changing, car-following are taken into account. The specific parameter in the proposed modelling and simulation is car-following under a consideration of agents. Diligent driver is also added to the proposed car-following parameter. The number of diligent drivers will be probabilistically determined. Although the proposed simulation is based on the Nagel-Schreckenberg traffic cellular automata, lane changing and new car-following parameters are specific to the proposed modelling and simulation.

3. Overview of the car-following models

There are two major methods of car-following, continuous and discrete models. Some of continuous models are based on the optimal velocity model (OVM), generalized force model (GFM), full velocity difference model (FVDM), and two velocity difference model (TVDM). Meanwhile, cellular automaton model is used for the discrete model.

3.1 Continuous models

In the micro traffic model, the drivers are stimulated their own velocity v_n, the distance between the car and the car ahead x_n, and the velocity of the vehicle in front $v_{n\text{-}1}$. The equation of vehicle motion is characterized by the acceleration function which depends on the input stimuli,

$$\ddot{x}_n(t) = \dot{v}_n(t) = F\left(v_n(t), x_n(t), v_{n-1}(t)\right) \tag{1}$$

3.1.1 Optimal Velocity Model (OVM)

The OVM is a dynamic model of traffic congestion based on a vehicle motion equation. In this model, the optimal velocity function of the headway of the preceding vehicle is introduced. Congestion may occur due to induce by a small perturbation without any specific origin such as a traffic accident or a traffic signal. The OVM can regard to this congestion phenomenon as the instability and the phase transition of a dynamical system (Bando, M., et al., 1995).

The vehicle motion equation is expressed with the assumption that each vehicle driver responds to a stimulus from other vehicles in some specific fashion. The response is followed by acceleration. The sensitivity of acceleration to stimulus is a function of the vehicle position, his time derivatives (or the distance), and so on. This function is determined by the drivers' characteristics, whether or not the vehicle drivers obey postulated traffic regulations at all times in order to avoid traffic accidents.

There are two major types of drivers. The first type is based on the idea that each vehicle must maintain the legal safe distance of the preceding vehicle, which depends on the relative velocity of these two successive vehicles. The second type is that each vehicle has the optimal velocity, which depends on the following distance of the preceding vehicle. In the OVM, the traffic dynamics equation is proposed based on the latter assumption results in a realistic traffic flow model. In this proposed model, the stimulus is assumed to be a function of a following distance and the sensitivity is a constant. The OVM ignores the vehicle length and assumes that all the drivers have common sensitivities. Then each vehicle has the optimal velocity V and that each vehicle driver responds to a stimulus from the vehicle ahead. Driver has to control their speed for maintaining the legal safe velocity in accordance with the motion of the preceding vehicle.

The dynamical equation of the system is obtained (based on the acceleration equation) as,

$$\frac{d^2x_n(t)}{dt^2} = a\left[V\left(\Delta x_n(t)\right) - \frac{dx_n(t)}{dt}\right] \tag{2}$$

where

$$\Delta x_n(t) = x_{n+1}(t) - x_n(t) \tag{3}$$

n denotes vehicle number (n = 1, 2, …, N). N is the total number of vehicles, a is a constant representing the driver's sensitivity (which has been assumed to be independent of n), and x_n is the location of the nth vehicle. The OVM assumes that the optimal velocity $V(\Delta x_n)$ of n^{th} vehicle depends on the distance between the n^{th} vehicle and the $(n\text{-}1)^{\text{th}}$ vehicle (preceding vehicle) (a distance-dependent optimal velocity). When the headway becomes short the velocity must be reduced and become small enough to avoid crash. On the other hand, when the headway becomes longer the driver accelerates under the speed limit, the maximum velocity. Thus, V is represented as to have the following properties,

i. a monotonically increasing function, and
ii. $\left|V(\Delta x)\right|$ has an upper bound.

$$V^{\max} \equiv V\left(\Delta x \to \infty\right).$$

The OVM takes the optimal velocity function $V(\Delta x_n)$ as,

$$V(\Delta x_n) = \tanh(\Delta x_n) \tag{4a}$$

$$V(\Delta x_n) = \tanh(\Delta x_n - 2) + \tanh 2 \tag{4b}$$

where Equation (4a) is called as the simple model and also Equation (4b) is called as the realistic model.

3.1.2 Generalized Force Model (GFM)

The driver behaviour is mainly given by the motivation to reach a certain desired velocity v_n (which will be reflected by an acceleration force), and by the motivation to keep a safe distance from other car (n-1) (which will be described by repulsive interaction forces). The GFM is created to improve the OVM which has the problems of too high acceleration and unrealistic deceleration (Helbing, D. & Tilch, B., 1998).

In the GFM, one term is added to the right-hand side of Equation (2). Thus the formula of the GFM is written by the following equation,

$$\frac{d^2x_n(t)}{dt^2} = a\left[V(\Delta x_n(t)) - \frac{dx_n(t)}{dt}\right] + \lambda\Theta(-\Delta v)(\Delta v) \tag{5}$$

where Θ denotes the Heaviside function, λ is a sensitivity coefficient different from a. Note that in the GFM, Equation (5) can be rewritten as follows,

$$\frac{d^2x_n(t)}{dt^2} = a\left[v_m - v_n(t)\right] + a\left[V(\Delta x_n(t)) - v_m\right] + \lambda\Theta(-\Delta v)(\Delta v) \tag{6}$$

where v_m is the maximum speed. The first term on the right-hand side is the acceleration force, and the last two terms represent the interaction forces.

3.1.3 Full Velocity Difference Model (FVDM)

A full velocity difference mode (FVDM) (Jiang, R., et al., 2001) for a car-following theory is based on the previous models. The FVDM model includes car-following parameter to the previously proposed models. Through numerical simulation, property of the model is investigated using both analytic and numerical methods. It was found that the FVDM model can represent the phase transition of traffic flow and also estimate the evolution of traffic congestion.

On the basis of the GFM formula, taking the positive Δv factor into account, the FVDM is described as the following dynamics equation,

$$\frac{d^2x_n(t)}{dt^2} = a\left[V(\Delta x_n(t)) - \frac{dx_n(t)}{dt}\right] + \lambda\Delta v \tag{7}$$

The FVDM takes both positive and negative velocity differences into account. The model equation of the FVDM (7) may be reformulated into the similar form in the Equation (8).

The GFM assumes that the positive Δv does not contribute to the vehicle interaction, while the FVDM suggests that it contributes to vehicle interaction by reducing interaction force because $a\left[V\left(\Delta x_n(t)\right)-v_m\right]$ is always negative and $\lambda\Theta(\Delta v)\Delta v$ is always positive.

$$\frac{d^2x_n(t)}{dt^2}=a\left[v_m-v_n(t)\right]+a\left[V\left(\Delta x_n(t)\right)-v_m\right] \\ +\lambda\Theta(-\Delta v)\Delta v+\lambda\Theta(\Delta v)\Delta v \tag{8}$$

One of the examples of the application of the FVDM on car motion simulation with traffic signal shows that it can describe the traffic dynamics most exactly so that the FVDM is verified as a reasonable and realistic model. On the other hand real situation exist in between the FVDM and the GFM. In the FVDM, car accelerates more quickly than the car in the GFM. Therefore, the delay time δt in FVDM is smaller than that in GFM as is shown in Table 1. Since the empirical deceleration and acceleration are restricted in between the range from −3 m/s² to 4 m/s²] (Helbing, D. & Tilch, B., 1998), the FVDM has too high deceleration.

Model	Delay time δt (s)
OVM (a = 0.85 s^{-1})	1.6
GFM (a = 0.41 s^{-1})	2.2
FVDM (a = 0.41 s^{-1})	1.4

Table 1. Delay times of car motions from a traffic signal and disturbance propagation speed at jam density in different models (source: [16])

3.1.4 Two Velocity Difference Model (TVDM)

Two velocity different models (TVDM) for a car following theory are then proposed taking navigation in modern traffic into account. The property of the model is investigated using linear and nonlinear analysis (Ge, X.H., et al., 2008).

Intelligent Transportation System (ITS) plays an important role in the rapid development of modern traffic. By using such navigation system, drivers can obtain the information that they need. In accordance with the above concept, on the basis of the OVM, taking both Δv_n and Δv_{n+1} into account, (Ge, X.H., et al., 2008) obtain a more useful model called the two velocity difference model (TVDM), the following dynamics equation is expressed,

$$\frac{d^2x_n(t)}{dt^2}=a\left[V\left(\Delta x_n(t)\right)-\frac{dx_n(t)}{dt}\right]+\lambda G(\Delta v_n,\Delta v_{n+1}) \tag{9}$$

where $G(.)$ is a generic, monotonically increasing function, and is assumed to be a linear form as,

$$G\left(\Delta v_n,\Delta v_{n+1}\right)=p\Delta v_n+(1-p)\Delta v_{n+1} \tag{10}$$

where p is weighting value. The proper value of p could be lead to desirable results.

3.2 Discrete model: cellular automata model

Cellular Automata (CA) is a model that is discrete in space, time and state variables. The latter property distinguishes CA e.g. from discretised differential equations. Due to the

discreteness, CA is extremely efficient in implementations on a computer. CA for traffic has been called by traffic cellular automata (TCA). Some of the TCA models, i.e. (1) Deterministic model: Wolfram's rule 184 (CA-184); (2) Stochastic model: Nagel–Schreckenberg TCA (STCA)

In 1992, Nagel and Schreckenberg proposed a TCA model that was able to reproduce several characteristics of real-life traffic flows, e.g., the spontaneous emergence of traffic jams (Nagel, K. & Schreckenberg, M., 1992). Their model is called the *NaSch TCA*, but is more commonly known as the stochastic traffic cellular automaton (STCA). It explicitly includes a stochastic noise term in one of its rules.

The computational model in the STCA is defined on a one-dimensional array of L sites and with open or periodic boundary conditions. Each site may either be occupied by one vehicle or it may be empty. Each vehicle has an integer velocity with values between zero to $v_{\max}$. For an arbitrary configuration, one update of the system consists of the following four consecutive steps, which are performed in parallel for all vehicles,

1. Acceleration

$$v_{(i,j)}(t-1) < v_{\max} \wedge gs(i,j)(t-1) > v_{(i,j)}(t-1)+1 \Rightarrow v_{(i,j)}(t) \leftarrow v_{(i,j)}(t-1)+1 \tag{11}$$

($gs_{(i,j)}(t)$ space gap at each time step t or the distance to the next vehicle ahead).

2. Braking

$$gs_{(i,j)}(t-1) \le v_{(i,j)}(t) \Rightarrow v_{(i,j)}(t) \leftarrow gs_{(i,j)}(t-1)-1 \tag{12}$$

3. Randomization

$$\xi(t) < p \Rightarrow v_{(i,j)}(t) \leftarrow \max[0, v_{(i,j)}(t)-1] \tag{13}$$

($\xi(t)$ is random number, p is stochastic noise parameter or slowdown probability).

4. Vehicle movement

$$x_{(i,j)}(t) \leftarrow x_{(i,j)}(t-1) + v_{(i,j)}(t) \tag{14}$$

Through the step one to four very general properties of single lane traffic are modelled on the basis of integer valued probabilistic cellular automaton rules. Already this simple model shows nontrivial and realistic behaviour. Step 3 is essential in simulating realistic traffic flow otherwise the dynamics is completely deterministic. It takes into account natural velocity fluctuations due to human behaviour or due to varying external conditions. Without this randomness, every initial configuration of vehicles and corresponding velocities reaches very quickly at a stationary pattern which is shifted backwards (i.e. opposite the vehicle motion) in one site per time step.

4. Case study on evacuation from disaster occurred area based on agent-based approach

4.1 Situation

There are some of subsystems in the proposed micro traffic agent-based modelling and simulation. First is determining of the shape of road structure. It is conducted for the realistic

situation in Sidoarjo hot mudflow disaster that the road structure is straight road, one-way direction, and has some of traffic lanes, as well as there is no traffic light over there. With regard to the unpredictable disturbance properties, it has constant speed and same direction with vehicle on unidirectional road. Besides that the disaster comes from the one of the ends of the road. These conditions are appropriated with the real condition of Sidoarjo hot mudflow.

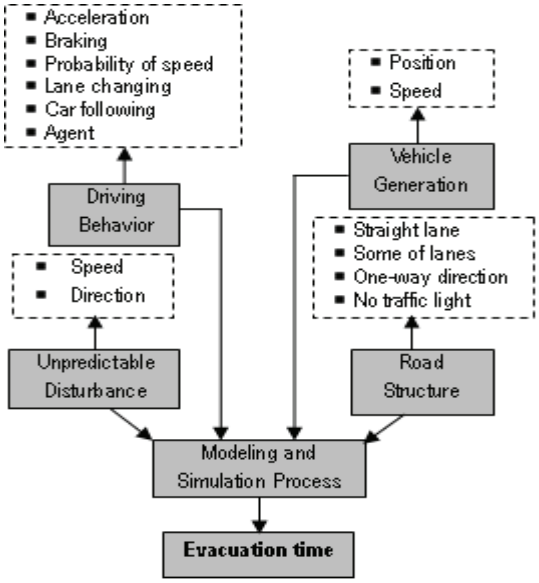


Fig. 1. Diagram block of evacuation simulation in the micro traffic

The other subsystem: vehicle generation, it is determined by a random number generation. It provides positions and speeds of all vehicles. Furthermore, we determine the driving behaviour. This study uses modified driving behaviour of Nagel-Schreckenberg proposed by using traffic Cellular Automata (Nagel, K. & Schreckenberg, M., 1992). We add two parameters in their model those are lane-changing and car-following. All the parameters of driving behaviour used in the proposed simulation are acceleration, braking, probability of speed, lane-changing, car-following, and agent. The overall simulation flow with the parameters used is shown in Fig. 1.

As the simulation results, we evaluated performance of the proposed driving behaviour (driving behaviour with agent) by compare it to the driving behaviour based on Nagel-Schreckenberg (without agent) for vehicle evacuation simulation from the affected disaster area of Sidoarjo. Thus, the simulation with and without agent cars is shown in this section.

4.2 Simulation procedure

The steps of proposed simulation model are preparation of road structure, vehicle generation, interpretation of unpredictable disturbance, and driving behaviour.

4.2.1 Preparation of road structure

The assumed road structure is shown in Fig.2. The realistic situation of main road structure is very close to mud containment walls (dikes) and the road shape is straight line. High

hazard will be occurred when the dike of hot mudflow is broken and the mud is spill over from the broken dike to the nearby roads spontaneously.

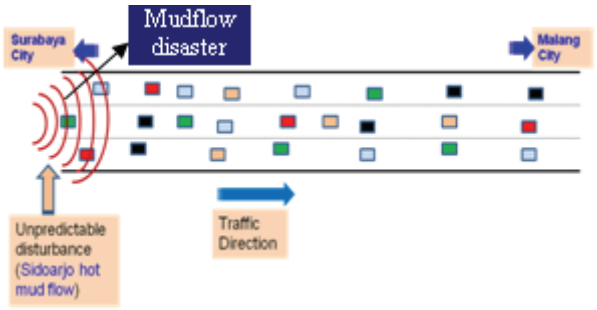


Fig. 2. Map of the roadway (Sidoarjo Porong roadway)

The hot mud will flow from behind of vehicles. It implies that the mudflow comes from the one of the ends of the road in the figure. Vehicles and hot mudflow have same direction (in Fig. 2, it is from left to right). Although the road is very close to the high dike of the hot mudflow, the transportation density on the road is also very high. This is the real situation that is main artery of traffic. The road has two lanes in one-way direction. The other actual condition on the road is that there is no traffic light at all.
In the proposed simulation, two lanes of traffic are assumed by condition above. Although the density is very high, drivers have a chance to change the lane.

4.2.2 Vehicle generation

The vehicle generation uses random number generator of Merssene Twister. Position and speed of vehicle depend on the probability of vehicle density.
Procedure for the determining of vehicle generation is as follows:
(1) Define number of lane (i = 1…k); (2) Define number of road length (j = 1… n); (3) Determine probability of vehicle density P_d; (4) Generate vehicles position $x(i, j)$ and their speed v_s randomly toward P_d

$$x_{(i,j)}(t) = [1 : v_{\max}] \tag{15}$$

with probability P_d.

4.2.3 Unpredictable disturbance

The proposed unpredictable disturbance is in the case of Sidoarjo hot mudflow disaster. It has two parameters, speed and direction. We assumed that speed of hot mudflow is to be constant. It is set as smaller than maximum speed of vehicle. Next the second parameter, direction of hot mudflow is the same as the vehicle's direction in the one-way street road.

4.2.4 Driving behavior

According to the agent, there are two driving behaviours, with and without agent cars. If an agent car exists, then the following cars recognize speed changes of the agent car so that traffic might be possible to control by the agent car as is shown in Fig.3. And if the agent car knows the best way to minimize the evacuation time (such information can be derived from

the evacuation control centre and transferred to the agent cars through wireless network connection), they could lead the following cars to the safe areas in a fastest way.

4.2.4.1 Modified Driving Behaviour of Nagel-Schreckenberg

Regarding to the Equation (11) to Equation (14), STCA have four steps of driving behaviour rule: acceleration, braking, randomization (slowdown probability), and vehicle movement, our study modifies it by adding two parameters about lane changing and car following. It is based on (Maerivoet, S. & De Moor, B., 2005 B) that stated the basic implementation of a lane-changing model in traffic cellular automata setting leads to two sub steps that are consecutively executed at each time step of the cellular automata. We called this modification is modified driving behaviour of Nagel-Schreckenberg.

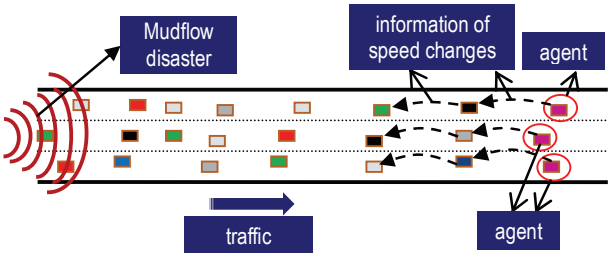


Fig. 3. Driving behaviour with agent

The overall rule of the modified driving behaviour of Nagel-Schreckenberg is as follows:

1. Acceleration

$$v_{(i,j)}(t-1) < v_{\max} \wedge gs(i,j)(t-1) > v_{(i,j)}(t-1)+1 \Rightarrow v_{(i,j)}(t) \leftarrow v_{(i,j)}(t-1)+1 \tag{16}$$

2. Braking

$$gs_{(i,j)}(t-1) \le v_{(i,j)}(t) \Rightarrow v_{(i,j)}(t) \leftarrow gs_{(i,j)}(t-1)-1 \tag{17}$$

3. Randomization

$$\xi(t) < p \Rightarrow v_{(i,j)}(t) \leftarrow \max\left[0, v_{(i,j)}(t)-1\right] \tag{18}$$

4. Vehicle movement

$$x_{(i,j)}(t) \leftarrow x_{(i,j)}(t-1)+v_{(i,j)}(t) \tag{19}$$

5. Lane changing
   Determine probability of lane changing P_{lc} and $a = [0 : v]$ for:

$$gs_{(i=1,j)}(t-1) < v \wedge x_{(i=2,j,j+v)}(t-1) = 0 \Rightarrow x_{(i=2,j+a)}(t) \leftarrow x_{(i=1,j)}(t-1) \tag{20}$$

   or

$$gs_{(i=2,j)}(t-1) < v \wedge x_{(i=1,j,j+v)}(t-1) = 0 \Rightarrow x_{(i=1,j+a)}(t) \leftarrow x_{(i=2,j)}(t-1) \tag{21}$$

6. Car following/vehicle movement: Back to step 4).

4.2.4.2 Proposed Driving Behaviour

There is the related work on the essence of the phenomenological research (Kretz, T., 2007). It stated that, (1) Concerning irrational behaviour: "After five decades studying scores of disasters such as floods, earthquakes and tornadoes, one of the strongest findings is that people rarely lose control."; (2) Concerning cooperation and altruism: "When danger arises, the rule as in normal situations is for people to help those next to them before they help themselves."; (3) Concerning *panic*: "Most survivors who were asked about panic said there was none."; and (4) Instead there were stories of people helping their spouses, flight attendants helping passengers, and strangers saving each other's lives.

Our proposed assumption for building up the agent rule in driving behaviour based on the statements above. When disaster occurs, every vehicle on affected road area has to have a good knowledge of driving behaviour. One of the important things in this situation is that all vehicles have a good capability of speed control followed by helping each other without any panic. The proposed assumption has the sense of a necessity for mimicking the basic features of real-life traffic flows in affected road area.

According to the ant behaviour (Retired Robots, In: http://www.ai.mit.edu/projects/ants/social-behavior), we make a technically driving behaviour. Agent behaviour can be built in some of vehicles. Each agent has appropriate information of speed control and is situated in each. Agent leads other vehicles so that traffic speed can be controlled by the agents. In this situation, the following car-following parameter is getting more important. If the following car does not follow the leading agent car, the traffic condition is worthless. This condition is consecutively performed to all vehicles in one lane and parallel to the entire lane.

We put the agent behaviour in the car-following parameter of driving behaviour. The rule of agent is,

$$\begin{aligned} x_{(i,j+v+c)}(t-1) = 0 \wedge v_{(i,j)}(t) + c \le v_{\max} \\ \Rightarrow x_{(i,j)}(t) \leftarrow x_{(i,j)}(t-1) + v_{(i,j)}(t) + c \end{aligned} \tag{22}$$

where c is positive integer.

Furthermore, the rule of proposed driving behaviour as follows:

1. Acceleration

$$\begin{aligned} v_{(i,j)}(t-1) < v_{\max} \wedge gs(i,j)(t-1) > v_{(i,j)}(t-1) + 1 \\ v_{(i,j)}(t) \leftarrow v_{(i,j)}(t-1) + 1 \end{aligned} \tag{23}$$

2. Braking

$$gs_{(i,j)}(t-1) \le v_{(i,j)}(t) \Rightarrow v_{(i,j)}(t) \leftarrow gs_{(i,j)}(t-1) - 1 \tag{24}$$

3. Randomization

$$\xi(t) < p \Rightarrow v_{(i,j)}(t) \leftarrow \max[0, v_{(i,j)}(t) - 1] \tag{25}$$

4. vehicle movement

$$x_{(i,j)}(t) \leftarrow x_{(i,j)}(t-1) + v_{(i,j)}(t) + c \tag{26}$$

5. Lane changing
   Determine probability of lane changing P_{lc} and $a = [0 : v]$ for:

$$gs_{(i=1,j)}(t-1) < v \wedge x_{(i=2,j,j+v)}(t-1) = 0 \Rightarrow x_{(i=2,j+a)}(t) \leftarrow x_{(i=1,j)}(t-1) \quad (27)$$

Or

$$gs_{(i=2,j)}(t-1) < v \wedge x_{(i=1,j,j+v)}(t-1) = 0 \Rightarrow x_{(i=1,j+a)}(t) \leftarrow x_{(i=2,j)}(t-1) \quad (28)$$

6. Car following/vehicle movement: Back to step 4).

4.3 Evaluation of the proposed parameter in the driving behaviour

Road traffic is always in a specific state that is characterized by three macroscopic variables: the flow rate q (cars per time step), the density k (cars per site), and the mean speed v (site per time step). Combination of all the possible homogeneous and stationary traffic states in an equilibrium function can be described graphically by three diagrams. The equilibrium relations presented in this way are known under the name of fundamental diagrams. The fundamental relation is,

$$q = kv \quad (29)$$

There are only two independent variables density k and mean speed v.

Related to the fundamental relation of three macroscopic variables, we have conducted the relationship between the parameters proposed in the car-following of driving behaviour and the evacuation time T. These proposed parameters are the mean speed v and the density function c, which c is defined by $1/k$, completely $c = Q(1/k)$, Q is a function. So there are two relationships: mean speed v versus evacuation time T (v-T diagram) and function of density c versus evacuation time T (c-T diagram).

Function of density c is equivalent to the parameter c in Equation (22), while mean speed v is $v_{(i,j)}(t)$. Thus, parameter c in Equation (22) is better known under the name of function of density (= $Q(1/k)$). So that Equation (22) can be rewritten as,

$$x_{(i,j+v+Q(1/k))}(t-1) = 0 \wedge \left(v_{(i,j)}(t) + Q(1/k)\right) \leq v_{\max} \Rightarrow x_{(i,j)}(t) \leftarrow x_{(i,j)}(t-1) + v_{(i,j)}(t) + Q(1/k) \quad (30)$$

where $c = Q(1/k)$.

The previous work (Immers, L.H. & Logghe, S., 2002); (Maerivoet, S. & De Moor, B., 2005 A); (Maerivoet, S. & De Moor, B., 2005 B) and (Tampère, C.M.J., 2004) observed the relation between density k and mean speed v in the fundamental diagram (k-v diagram). There are also some special state points that require extra attention. One of them is about saturated traffic. On saturated roads, flow rate q and speed v are down to zero. The vehicles are queuing and there is a maximum density $k_{\max}$ (jam density). We can say on the other hand about the aforementioned special state point generally that by the density k increase (pass through the jam density), then the speed v will be decrease (down to zero). This condition is consistent with the characteristic of k-v diagram.

Based on k-v diagram, we can say in accordance with relation between mean speed v and evacuation time T (v-T diagram) that when the mean speed v down to zero (in the time the

density k has increased/pass the jam density), we found the evacuation time T by the great value. Besides, we also observe that by the increase of the mean speed v, we find the evacuation time T decreases. It occurs either with or without agent in the evacuation simulation (see Fig.4 and Fig.5 with/without agent respectively).

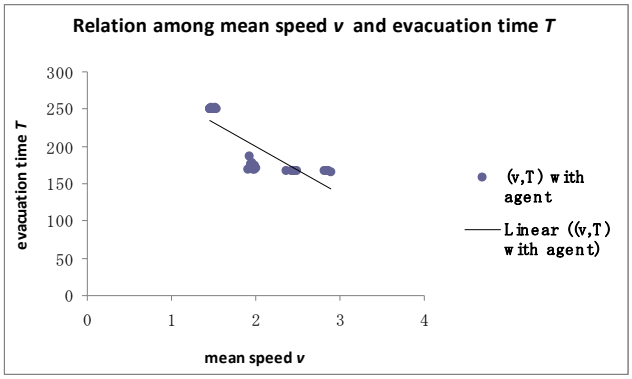


Fig. 4. Relationship between mean speed and evacuation time (with agent)

According to k-v diagram and relation between mean speed v versus evacuation time T (v-T diagram), we saw sequentially that by the increase of density k, mean speed v will decreases. And then, we also found that the movement of the mean speed v down to smaller value has impact the evacuation time T is going up. On the other hand, we have the sense of relation between the density k and the evacuation time T. Both k and T have linear correlation. When the density k goes up, the evacuation time is also going up.

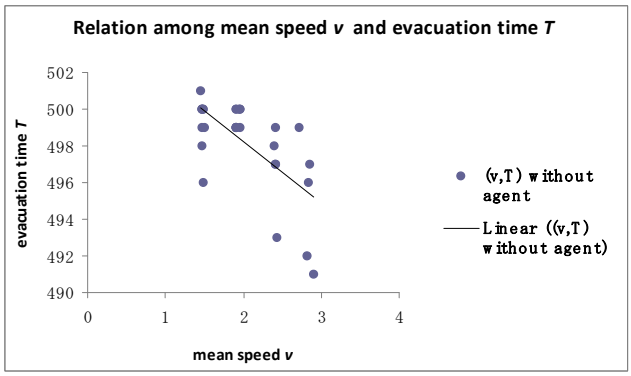


Fig. 5. Relationship between mean speed and evacuation time (without agent)

We knew that function of density c (= $Q(1/k)$) is in inverse ratio by k. It has the meaning that by using the density k larger, the function of density c has small value. While the density k and the evacuation time T have linear correlation then we say that in the time c has the small value, the value of the evacuation time T is large. The other hand, when the value of c is going up the evacuation time T will down to the smaller value. Our experiment results about value of c and T found relationship between both of them. Based on linear correlation

between the evacuation time T and the density k, we looked for function of density c (= $Q(1/k)$). From this correlation we obtained the value of $c = Q(1/k) = (12/10)(1/k)$. By using this value of c, we have the experiment results, relation between function of density c and evacuation time T (see Fig. 6). The pattern of their relationship in accordance with the description above. We showed it with/without agent.

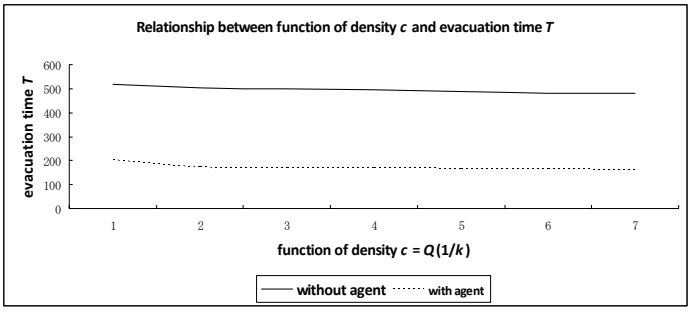


Fig. 6. Relationship between function of density c and evacuation time (with/without agent)

5. Survey results of traffic on the Sudioarjo Porong roadway

5.1 Specification of survey

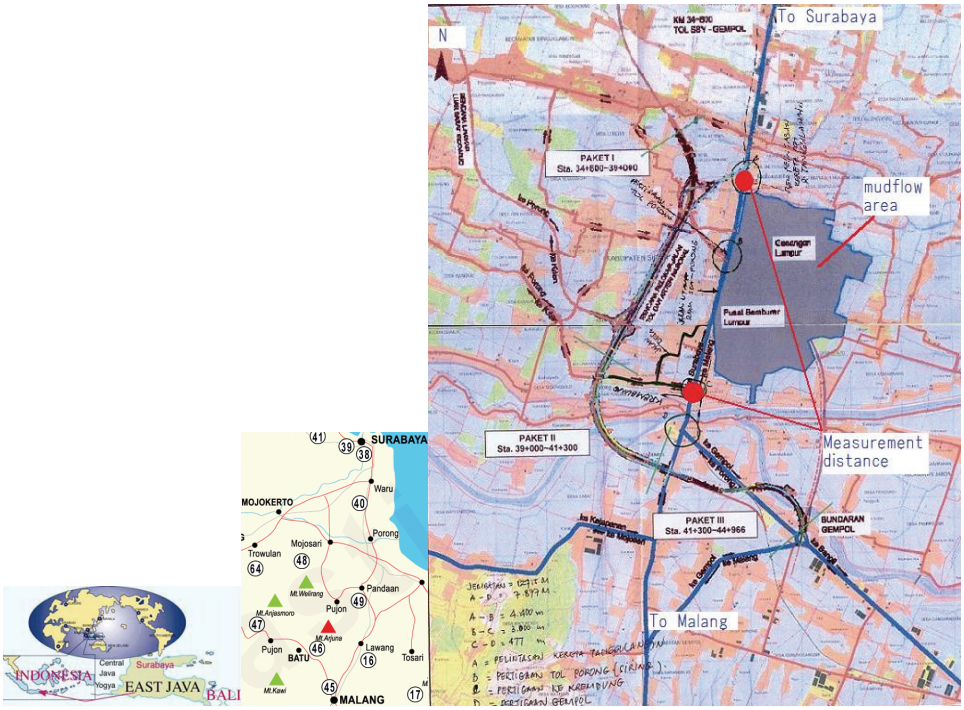
Roadway: Sidoarjo Porong roadway, East Java, Indonesia (Fig.7); direction of traffic: Sidoarjo/Surabaya to Malang/Banyuwangi (unidirectional); parameter was observed in the survey: speed of vehicle; data measurement: speed of vehicles every 15 minutes, during 24 hours, along eight days consecutively. All vehicles passing through the Sidoarjo Porong roadway is classified into four types: bus, truck/trailer, public transport, and private car.

5.2 The influence of cars speed with respect to the driving behavior

During survey the traffic data, we have totally 190 data of speed for each kind of cars (bus, truck/trailer, public transport, and private car). Analysis data is done to get any information related with the driving behavior.

Based on statistical hypothesis testing (t-student distribution), we find the hypothesis results for bus, public transport, and private car are accepted, whereas truck/trailer is rejected. It means that truck/trailer has different speed behaviour if it is compared by the other cars (bus, public transport, and private car). Speed of the truck/trailer is lower than that of the others. Comparison of speed between truck/trailer and all kind of the cars (mean speed of all vehicles) is conducted. Many trucks/trailers have the speed is less than or equal to 40 kilometre per hour (km/hr) when we compare it with all kind of the cars per speed interval, while all kind of the cars have the speed more than 41 km/hr is larger than that of the truck/trailer (see in Table 2 and Fig.8).

We also find distribution of speed for the bus, public transport, and private car. Comparison of their speeds related to the speed of all kind of the cars is conducted. Bus and public transport have the most frequency in the range of speed 36 – 40 km/hr, while for the private cars in the range 41 – 45 km/hr. We also found that all kind of the cars have the most frequency in the range of speed 41 – 45 km/hr (see in Table 3 and Fig.9).



(a) East Java (b) Surabaya to Malang (c) Measured roadway

Fig. 7. The roadway where the traffic survey is conducted

Back to the speed of the trucks/trailers, we show their speeds compared by the speed of all kind of the cars by time to time consecutively in 190 hours (eight days). We find that on the same measurement time, speeds of the trucks/trailers dominantly are lower than that of speed all kind of the cars (see in Fig. 10), number of lower speeds for the truck/trailer is around 81%.

Speed Interval	Truck	All	Speed Interval	Truck	All
1-5	13	11	36-40	52	43
6-10	2	4	41-45	35	51
11-15	1	1	46-50	2	20
16-20	3	2	51-55	0	0
21-25	19	10	56-60	0	0
26-30	27	24	61-65	0	0
31-35	36	24	66-70	0	0
			Total	190	190

Table 2. Distribution of speed for the truck/trailer and all kind of the cars

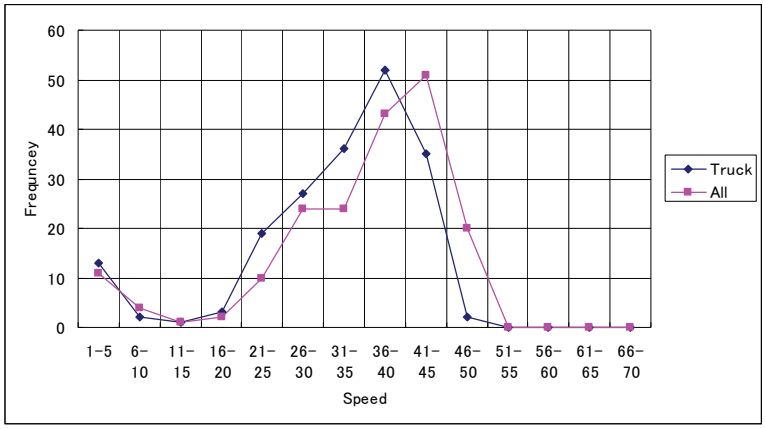


Fig. 8. Distribution of speed for the truck/trailer and all kind of the cars

Speed Interval	Bus	Public	Private	All	Speed Interval	Bus	Public	Private	All
1-5	11	11	12	11	36-40	51	40	30	43
6-10	4	4	3	4	41-45	37	39	36	51
11-15	2	1	1	1	46-50	24	27	31	20
16-20	3	1	2	2	51-55	1	3	21	0
21-25	4	7	11	10	56-60	0	1	3	0
26-30	28	25	20	24	61-65	0	0	0	0
31-35	24	31	19	24	66-70	1	0	1	0
					Total	190	190	190	190

Table 3. Distributions of speed for the bus; public transport; private car; and all kind of the cars

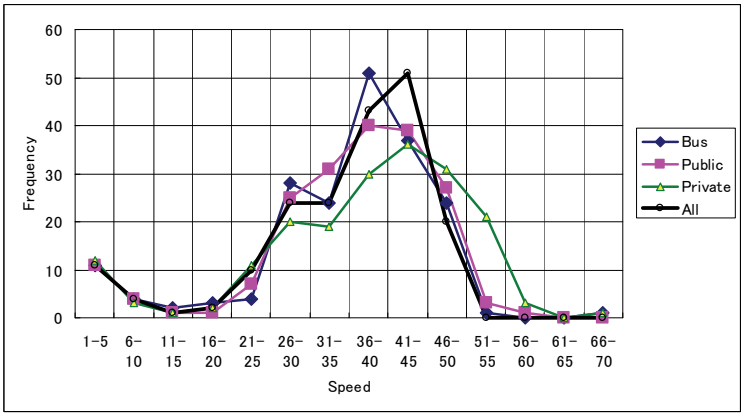


Fig. 9. Distributions of speed for the bus; public transport; private car; and all kind of the cars

Based on data analysis of speed for all vehicles, phenomena of traffics on the Sidoarjo Porong roadway related to the speed of vehicles was observed. We compare the speed each other of all the vehicle types and classify it. In general, number of trucks/trailers have the speed less than or equal to the speed interval (36 – 40 km/hr) is bigger than that the other vehicle types. On the other hand, is founded that the other types (bus, public transport, and private car) have the bigger number of cars available in the speed interval (41 – 45 km/hr) or more than that the trucks/trailers. Finally, according to the driving behaviour, we assume that there are diligent driver and usual driver regarding to their speeds. Dominantly, trucks/trailers become usual driver, while bus; public transport; and private car as diligent drivers.

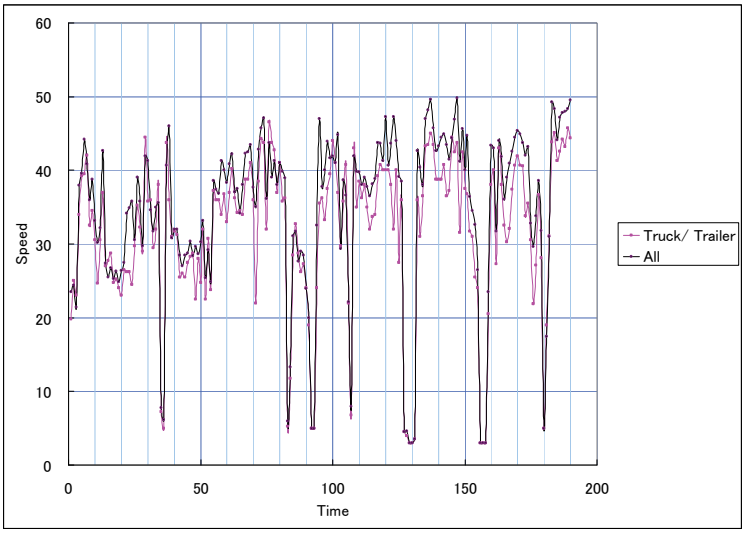


Fig. 10. Speed of the trucks and all kind of the cars in 190 hours (8days) respectively.

6. Modified car-following NaSch model with diligent drivers behaviour

Modified car-following Nagel-Schreckenberg (NaSch) model, is described by means of NaSch model. The rule set of modified car-following NaSch model have the same steps with the rule set of NaSch model for the first to third steps, They are acceleration, braking, and randomization, in Equation (11) to Equation (13). Next step in the NaSch model is vehicle movement shown in Equation (14). The NaSch model is implemented for one lane of traffic (road length is analogized by one-dimensional array L sites). In accordance to the road condition, modified car-following NaSch model uses two lanes of traffic (multi-lane traffic), it has consecutively lane-changing and vehicle movement for next steps. Due to the case of evacuation is reflected on the condition of Sidoarjo Porong roadway, modified car-following NaSch model concerns on the unidirectional road and no looping of traffic, while the NaSch model uses unidirectional road with looping of traffic (periodic boundary conditions).
This section, we propose another driving behavior in the modified car-following NaSch model other than agent driver, it is about diligent driver. Agent driver has been proposed in section 4.2.4.2 inserted into the driving behaviour of NaSch. Based on both of them, agent driver and diligent one, NaSch model is modified and called modified car-following NaSch

model. The number of diligent driver is probabilistically determined, while the agent driver is determined by integer. Both diligent driver and agent driver are reflected by the addition of speed in the car-following model. A diligent driver has the additional speed $c' = \left[0 : \min(\overline{v}, v)\right]$ while an agent has $c = \left[0 : \max(v)\right]$. We know that velocity v of each car is given by the function of the headway to the vehicle in front. By adding the additional speed to the diligent driver and agent driver then we have the velocity v,

$$v = \begin{cases} v_{(i,j)}(t) + c' = v_{(i,j)}(t) + \left[0 : \min(\overline{v}, v)\right] \\ for \quad diligent \quad driver \\ \\ v_{(i,j)}(t) + c = v_{(i,j)}(t) + \left[0 : \max(v)\right] \\ for \quad agent \quad driver \end{cases} \tag{31}$$

By referring (Maerivoet, S. & De Moor, B., 2005 B), we make the implementation of lane-changing model in the modified car-following NaSch model. Implicitly, we have used it in the section 4.2.4.1. There are two sub-steps on the lane-changing section that is consecutively executed at each time step. First, the lane-changing model is executed, exchanging vehicles between laterally adjacent lanes; second, all vehicles are moved forward by applying the modified car-following part of the NaSch model's rules. We make the steps of lane-changing as follows:

1. Lane-changing model

For two lanes of traffic, we proposed as follows:

Determine probability of lane changing P_{lc} and $a = [0 : v]$ for

$$\begin{aligned} gs_{(i=1,j)}(t-1) < v \wedge x_{(i=2,j,j+v)}(t-1) = 0 \\ \Rightarrow x_{(i=2,j+a)}(t) \leftarrow x_{(i=1,j)}(t-1) \end{aligned} \tag{32}$$

and

$$\begin{aligned} gs_{(i=2,j)}(t-1) < v \wedge x_{(i=1,j,j+v)}(t-1) = 0 \\ \Rightarrow x_{(i=1,j+a)}(t) \leftarrow x_{(i=2,j)}(t-1) \end{aligned} \tag{33}$$

The rules in Equation (32) and Equation (33) describe that if in a lane, the driver is not possible to move his car forward (there is car ahead) and he sees that there is any safety space in another lane with the number of space is up to the speed v then he changes the lane. If the car was already stayed in the new lane, then he has the speed less than or equal to the current speed v. It implies any deceleration experienced by the car when moving to the other lane by probability P_{lc}. Adjustment the number of probability P_{lc} obtains how many cars will change their lane when the condition is fulfilled.

2. Vehicle movement

For a diligent driver, his vehicle movement is:

$$x_{(i,j)}(t) \leftarrow x_{(i,j)}(t-1) + v_{(i,j)}(t) + [0 : \min(\overline{v}, v)] \tag{34}$$

with $gs(t\text{-}1) > v(t - 1)$.

While an agent driver:

$$x_{(i,j)}(t) \leftarrow x_{(i,j)}(t-1) + v_{(i,j)}(t) + [0 : \max(v)] \tag{35}$$

with $gs(t\text{-}1) > v(t - 1)$.

The rule of vehicle movement in the NaSch model is stated in the Equation (14). In this model the current position of the car $x(t)$ is influenced by the previously position $x(t\text{-}1)$ and the current speed $v(t)$. By reflection on the evacuation of vehicle on the suffered road (Sidoarjo Porong roadway), we propose diligent driver into the car-following. He has the speed is shown in Equation (31). Besides he has the current speed $v(t)$, he also has the additional speed between zero to $\min(\overline{v}, v)$ because diligent driver arranges his speed on the temporal average speed. Thus vehicle movement in the modified car-following NaSch model for a diligent driver has equation that is shown in Equation (34), the current position of the car $x(t)$ is not only influenced by the previously position $x(t\text{-}1)$ and the current speed $v(t)$ but also influenced by the diligent driver's additional speed. The other hand, an agent driver into this study has the speed is also expressed in Equation (31), he has the additional speed between zero to $\max(v)$ because agent driver arranges his speed to the maximum speed when the condition is available. Vehicle movement in the modified car-following NaSch model for an agent driver has equation that is expressed in Equation (35), the current position of the car $x(t)$ is influenced by the previously position $x(t\text{-}1)$, the current speed $v(t)$, and the agent driver's additional speed.

Lane changing causes the position of that car move forward in the new lane so that there is difference between the prior positions of the car (there is no moving forward in last lane) and the current position (moving forward by his speed in the new lane). This event provides the time of the car is faster to arrive in any safety areas. The accumulation of lane-changing will support to the effectiveness of vehicle evacuation. Vehicle movement for diligent driver and agent driver also has the important role to find the effectiveness of vehicle evacuation, especially on the suffered road, Sidoarjo Porong roadway.

A modified car-following NaSch model considering agent driver and diligent driver is presented as follows:

1. Acceleration:

$$\begin{aligned} v_{(i,j)}(t-1) < v_{\max} \wedge gs(i,j)(t-1) > v_{(i,j)}(t-1) + 1 \\ v_{(i,j)}(t) \leftarrow v_{(i,j)}(t-1) + 1 \end{aligned} \tag{36}$$

2. Braking:

$$gs_{(i,j)}(t-1) \le v_{(i,j)}(t) \Rightarrow v_{(i,j)}(t) \leftarrow gs_{(i,j)}(t-1) - 1 \tag{37}$$

3. Randomization:

$$\xi(t) < p \Rightarrow v_{(i,j)}(t) \leftarrow \max[0, v_{(i,j)}(t) - 1] \tag{38}$$

4. Vehicle movement: If a diligent driver,

$$x_{(i,j)}(t) = x_{(i,j)}(t-1) + v_{(i,j)}(t) + [0 : \min(\overline{v}, v)] \tag{39}$$

Else if an agent driver,

$$x_{(i,j)}(t) = x_{(i,j)}(t-1) + v_{(i,j)}(t) + [0 : \max(v)] \tag{40}$$

5. Lane-changing:
   Determine probability of lane changing P_{lc} and $a = [0 : v]$ for:

$$gs_{(i=1,j)}(t-1) < v \wedge x_{(i=2,j,j+v)}(t-1) = 0 \Rightarrow x_{(i=2,j+a)}(t) \leftarrow x_{(i=1,j)}(t-1) \tag{41}$$

   or

$$gs_{(i=2,j)}(t-1) < v \wedge x_{(i=1,j,j+v)}(t-1) = 0 \Rightarrow x_{(i=1,j+a)}(t) \leftarrow x_{(i=2,j)}(t-1) \tag{42}$$

6. Car-following/vehicle movement: back to step 4).

6.1 Experimental simulation results

The computational simulation of modified car-following NaSch model is carried out and compared with the results of the NaSch model in the case of evacuation. We observe the evacuation time with respect to the diligent driver and also the agent driver. The maximum speed is set to $v_{\max} = 5$ and the system size is $L = 200$.

6.1.1 Fundamental diagram and spatio-temporal structures

Fig.11 shows the fundamental diagram of modified car-following NaSch model. When the density k is lower than the critical density, the flow rate increases with k and the traffic flow is free. When k is larger than the critical density, the flow rate decreases as k increases and the traffic flow is congested. This situation is in accordance with traffic flow stated by (Immers, L.H. & Logghe, S., 2002); (Maerivoet, S. & De Moor, B., 2005 A); (Maerivoet, S. & De Moor, B., 2005 B) and (Tampère, C.M.J., 2004). In those references also state that at saturated roads (large density), the flow rate q is saturated (the vehicles are queuing). The condition is different with fundamental diagram resulted by modified car-following NaSch model (Fig. 11). When density is large, saturation of flow rate q does not happen; it is caused by the role of diligent driver and agent driver.

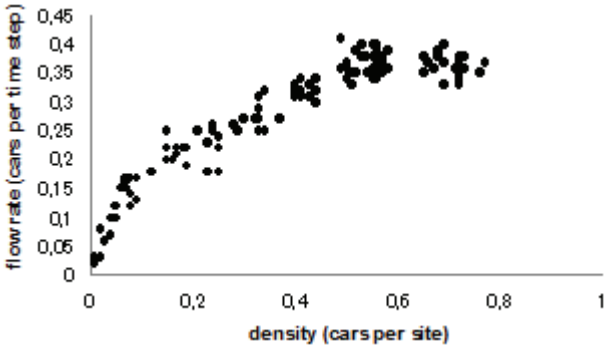


Fig. 11. Fundamental diagram of modified car-following NaSch model

Fig. 12 compares the spatio-temporal structures of the NaSch model and the modified car-following NaSch model. One can see that the evacuation time of the modified car-following

NaSch model is different from those of the NaSch model (previous model). As the number of agent increases in the modified car-following NaSch model, the evacuation time will decreases. Experimental simulation results using diligent driver 0.7; lane-changing 0.4; and density 0.6 (Fig. 12.), provide the evacuation time of the NaSch model T = 107, while the modified car-following NaSch model by using agent = 1 has T = 101; agent = 2, it has T = 95; and using agent = 3, T = 90. The existing of agent driver influenced reduction of the evacuation time T. For these experimental simulation results, by the agent driver three, there is difference of percentage ratio the evacuation time 16% decrease than that in the NaSch model (without agent driver). Traffic jams occur both in the NaSch model and modified car-following NaSch model. The black areas in Fig. 12 show the happening of traffic jams. Either in the modified car-following NaSch model or in the NaSch model, traffic jams emerge in uncertain time.

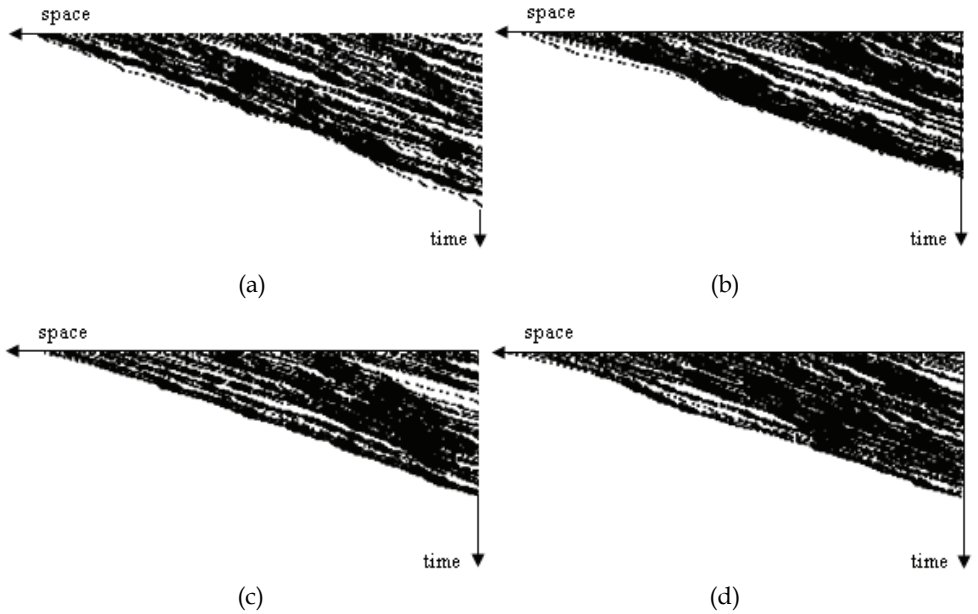


Fig. 12. Spatiotemporal diagrams of the previous model: NaSch model (a) and the modified car-following NaSch model (b) (c) (d) with different the number of agent. The vehicles drive from left to right. The vertical direction (down) is (increasing) evacuation time. (b) agent = 1; (c) agent = 2; (d) agent = 3. The parameters diligent driver = 0.7, lane-changing = 0.4, density k = 0.6.

6.1.2 Effect agent and diligent driver on the evacuation time

We observe the influence of agent and diligent driver with respect to the evacuation time based on lane-changing; mean speed; and diligent driver itself with different number of agent. Fig. 13 compares the effect of lane-changing in the NaSch model and the modified car-following NaSch model. By using lane-changing 0 to 0.6 and diligent driver 0.3; 0.5; 0.7 (for Fig. 13 (a); (b); (c) consecutively) we get the evacuation time in the modified car-following NaSch model is lower than that in the NaSch model. As lane-changing

increases, evacuation time will decreases not only in the modified car-following NaSch model but also in the NaSch model (previous model). In Fig. 13, we also note that the evacuation time more decreases when the number of agent is larger. Experimental simulation results in Fig. 13 also show that by the lane-changing increases, it will be found the evacuation time decreases.

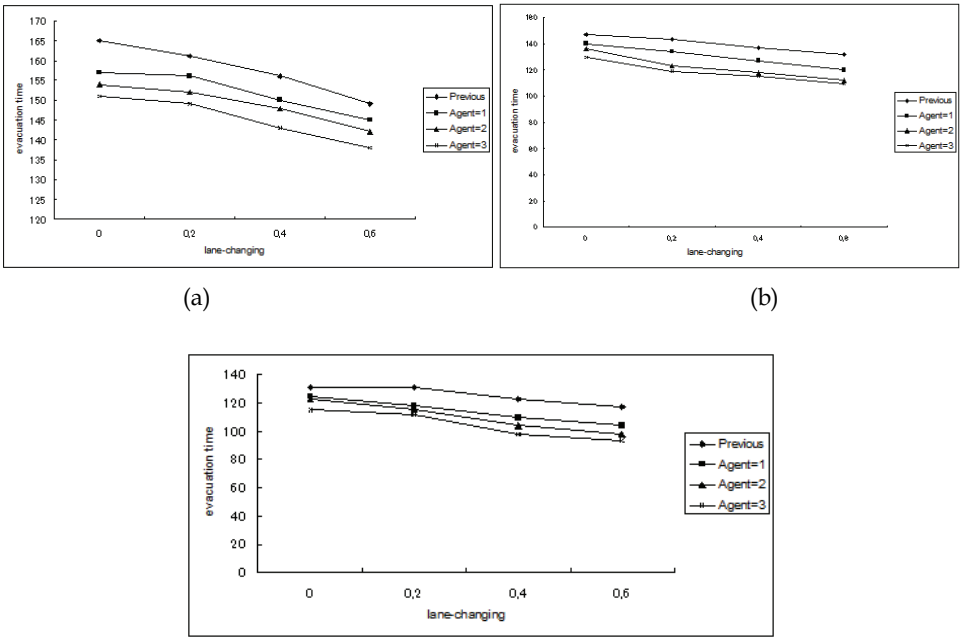
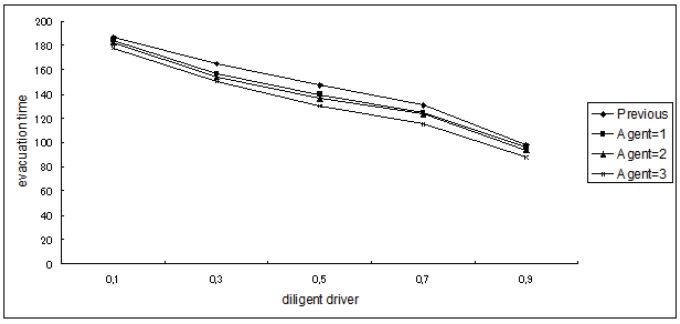


Fig. 13. The lane-changing evacuation time in the previous model: NaSch model and the modified car-following NaSch model. The density k = 0.6. (a) diligent driver = 0.3; (b) diligent driver = 0.5; (c) diligent driver = 0.7.

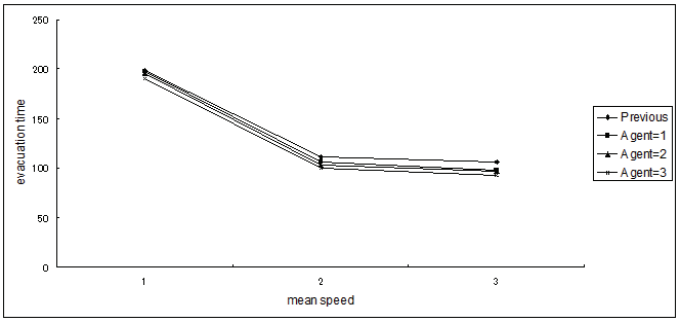
Furthermore, the effect of diligent driver and mean speed respectively with respect to the evacuation time is conducted. Fig. 14 (a) shows comparison of the effect of diligent driver between previous model (NaSch model) and modified car-following NaSch model. When the diligent driver increases, then either the evacuation time in the modified car-following NaSch model or in the NaSch model decrease. We find that it is lower in the modified car-following NaSch model than that in the NaSch model. These conditions occur not only in the agent = 1 but also in the agent = 2 and 3. Fig. 14 (a) also gives information that as the number of agent increase, the evacuation time will decreases.

The effect of modified car-following NaSch model with respect to the evacuation time based on mean speed is obtained in Fig.14 (b). As the mean speed increases, the evacuation time will decreases not only in the NaSch model but also in the modified car-following NaSch model. We note that the evacuation time in the modified car-following NaSch model is lower than that in the NaSch model when the mean speed increases. For the agent increases, we get the evacuation time in the modified car-following NaSch model decreases. Fig.14 (b)

also shows a leap of evacuation time from mean speed = 1 to mean speed = 2. It has 44% decreasing in the NaSch model and 47% in the modified car-following NaSch model.
We explain above (Fig.14 (a)) the influence of the modified car-following NaSch model with respect to the evacuation time based on diligent driver without lane-changing. Furthermore, we combine the diligent driver and lane-changing to get the evacuation time. Fig.15 expresses that by using lane-changing = 0.2 and diligent driver increases, we get the evacuation time decreases. It occurs both in the NaSch model and the modified car-following NaSch model. Evacuation time in the NaSch model is larger than that in the modified car-following NaSch model. We note that by the increase of agent driver in the modified car-following NaSch model, the evacuation time decreases.



(a)



b)

Fig. 14. Effect of Car-following model in the previous model: NaSch model and the modified car-following NaSch model based on (a) diligent driver, (b) mean speed. The density k = 0.6, lane-changing = 0.

Table 4 shows the effectiveness of the agent and diligent driver in terms of evacuation time. As the diligent driver increases, we have the percentage ratio of the effectiveness also increases in the modified car-following NaSch model (by agent = 1 and 3). We see that by using agent driver is three, the effectiveness is larger than that by using agent driver is one. Table 4 also describes that effect of diligent driver is almost double when the percentage ratio of diligent driver is 100%; i.e. by agent = 1, the effectiveness is 44%; while using agent = 3, it is 50%. Still in the percentage ratio of diligent driver is 100%, we also find the effect of

agent driver is also almost double when the number of agent driver is three in comparison to the existing simulation result without any agent (NaSch model) (in the NaSch model, the evacuation time = 183; while for the modified car-following NaSch model = 91).

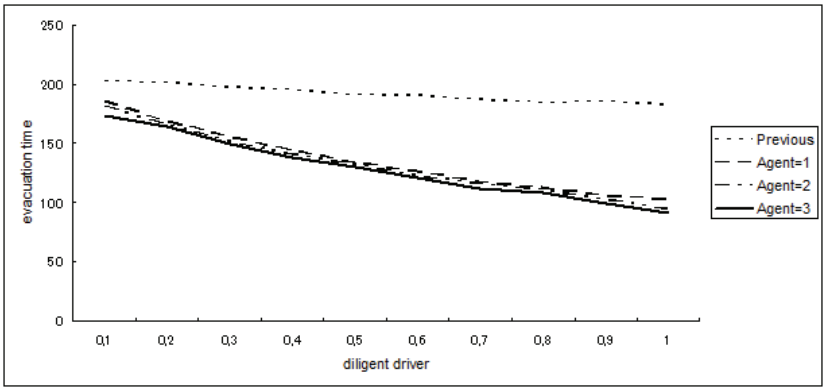


Fig. 15. The diligent driver evacuation time in the previous model: NaSch model and the modified car-following model. The density k = 0.6, lane-changing = 0.2.

dd (%)	Evacuation time				
	Previous model	Modified (*A*=1)	Effectiveness (%)	Modified (*A*=3)	Effectiveness (%)
10	202	186	8	173	14
20	201	169	16	164	18
30	197	156	21	149	24
40	195	144	26	138	29
50	191	134	30	130	32
60	190	126	34	120	37
70	187	118	37	112	40
80	184	113	39	108	41
90	185	106	43	99	46
100	183	102	44	91	50

Table 4. The effectiveness of agent and diligent driver in terms of evacuation time. The density 0.6 and lane-changing 0.2 (*dd*: diligent driver, *A*: the number of agent)

7. Conclusions

Agent and diligent driver are incorporated into the car-following of NaSch model. The modified car-following NaSch model is proposed. Evaluation of the proposed parameter, the fundamental diagram, spatio-temporal patterns, effect of lane-changing and car-following with respect to the evacuation time, combination parameter of diligent and agent driver in the case of evacuation time and the effectiveness are investigated.

The comparative simulation study between with and without agent as well as diligent drivers is conducted based on the NaSch model. In the modified car-following NaSch model, the effect of lane-changing and car-following towards the evacuation time are larger than that in the NaSch model. The simulation results show that the effect of diligent driver depends on the percentage ratio of diligent driver and is almost double when the percentage ratio of diligent driver is 100%. It is found that the effect of agent driver also depends on the number of agent driver and is almost double when the number of agent driver is three in comparison to the existing simulation result without any agent (NaSch model).

8. References

Bando, M., Hasebe, K., Nakayama, A., Shibata, A. & Sugiyama, Y. (1995). Dynamical model of traffic congestion and numerical simulation, *Physical Review E*, Vol. 51, Issue 2, (February 1995) 1035-1042, 1550-2376

Chen, X. & Zhan, F.B. (2008). Agent-based modelling and simulation of urban evacuation: relative effectiveness of simultaneous and staged evacuation strategies. *Journal of the Operational Research Society*, Vol. 59, No. 1, (January 2008) 25–33, 1476-9360

Church, R.L. & Sexton, R. (2002). Modelling small area evacuation: Can existing transportation infrastructure impede public safety?. *Final Report, Vehicle Intelligence & Transportation Analysis Laboratory*, University of California, Santa Barbara

Cova, T.J. & Church, R.L. (1997). Modelling community evacuation vulnerability using GIS. *International Journal of Geographical Information Science*, Vol. 11, No. 8, (December 1997) 763-784, 1365-8824

Cova ,T.J. & Johnson, J.P. (2002). Microsimulation of neighborhood evacuations in the urban-wildland interface. *Environment and Planning A*, Vol. 34, No. 12, 2211–2229, 1472-3409

Deadman, P.J. (1999). Modelling individual behaviour and group performance in an intelligent agent-based simulation of the tragedy of the commons. *Journal of Environmental Management*, Vol. 56, Issue 3, (July 1999) 159–172, 0301-4797

Ge, X.H., Cheng, R.J. & Li, Z.P. (2008). Two velocity difference model for a car following theory. *Physica A: Statistical Mechanics and Its Applications*, Vol. 387, Issue 21, (September 2008) 5239-5245, 0378-4371

Helbing, D. & Tilch, B. (1998). Generalized force model of traffic dynamics. *Physical Review E*, Vol. 58, Issue 1, (July 1998) 133-138, 1550-2376

Hobeika, A.G. & Jamei, B. (1985). Massvac: A model for calculating evacuation times under natural disaster. *Proceedings of the Conference on Emergency Planning, Simulation Series, Vo. 15, No. 1*, pp. 23–28, La Jolla, CA, January 1985, San Diego, CA

Immers, L.H. & Logghe, S. (2002). Traffic Flow Theory. Faculty of Engineering, Department of Civil Engineering, Section Traffic and Infrastructure, Kasteelpark Arenberg 40, B-3001 Heverlee, Belgium

Indahnesia.com (2007). Main toll road near Porong in southern direction closed. In: http://blog.indahnesia.com/entry/200704110832/main_tollroad_near_porong_in_southern_direction_closed.php, posted in Sidoarjo mudflow @ 11 April 2007 08:32 CET by Jeroen

Jiang, R., Wu, Q.S. & Zhu, Z.J. (2001). Full velocity difference model for a car-following theory, *Physical Review E*, Vol. 64, Issue 1, (June 2001) 017101-1 - 017101-4, 1550-2376

Kretz, T. (2007). Pedestrian Traffic, Simulation and Experiment. PhD Thesis, Vom Fachbereich Physik der Duisburg-Essen University

Maerivoet, S. & De Moor, B. (2005A). Traffic Flow Theory. *SISTA Internal Report 05-154*, ESAT-SCD, K.U.Leuven, Belgium, (July 2005)

Maerivoet, S. & De Moor, B. (2005B). Cellular automata models of road traffic. *Physics Reports*, Vol. 419, Issue 1, (November 2005) 1-64, 0370-1573

Mediacenter (2007). Mudflow Sidoarjo Porong East-Java Indonesia. In: http://mudflow-sidoarjo.110mb.com/lumpur23.htm

Nagel, K. & Schreckenberg, M. (1992). A cellular automaton model for freeway traffic. *Journal Physics I France*, Vol. 2, No. 12, (December 1992) 2221-2229, 1155-4304

Nagel, K. (1996). Particle hopping models and traffic flow theory. *Physical Review E*, Vol. 53, No. 5, (May 1996) 4655–4672, 1550-2376

Pidd, M., de Silva, F.N. & Eglese, R.W. (1996). A simulation model for emergency evacuation. *European Journal of Operational Research*, Vol. 90, Issue 3, (May 1996) 413-419, 0377-2217

Retired Robots. The Ants: A Community of Microrobots. Social Behaviour, Clustering Around Food. In: http://www.ai.mit.edu/projects/ants/social-behavior

Rifai, R. (2008). Spatial Modelling and Risk Assessment of Sidoarjo Mud Volcanic Flow. Master Thesis, Academic Output AES MSc theses 2008, Repository ITC publications, Faculty of Geo-Information Science and Earth Observation, University of Twente

Sheffi, Y., Mahmassani, H. & Powell, W.B. (1982). A transportation network evacuation model. *Transportation Research Part A: General*, Vol. 16, Issue 3, (May 1982) 209–218, 0965-8564

Stern, E. & Sinuany-Stern, Z. (1989). A behavioural-based simulation model for urban evacuation. *Papers in Regional Science*, Vol. 66, No. 1, (December 1989) 87–103, 1435-5957

Sugiman, T. & Misumi, J. (1988). Development of a new evacuation method for emergencies: Control of collective behaviour by emergent small groups. *Journal of Applied Psychology*, Vol. 73, Issue 1, (February 1988) 3–10, 1939-1854

Tampère, C.M.J. (2004). *Human-Kinetic Multiclass Traffic Flow Theory and Modelling: With Application to Advanced Driver Assistance Systems in Congestion*. TRAIL Research School , ISBN 90-5584-060-2, P.O. Box 5017, 2600 GA Delft, The Netherlands, T +31 15 278 60 46, F +31 15 278 43 33, E info@rsTRAIL.nl, I www.rsTRAIL.nl

Teodorovic, D.A. (2003). Transport modelling by multi-agent systems: A swarm intelligence approach. *Transportation Planning and Technology*, Vol. 26, Issue 4, (August 2003) 289–312, 1029-0354

10

Cellular Automata for Bus Dynamics

Ding-wei Huang[1] and Wei-neng Huang[2]
Department of Physics, Chung Yuan Christian University
Taiwan

1. Introduction

Recently, traffic dynamics has attracted much attention from physicists (Maerivoet & De Moor, 2005; Chowdhury et al., 2000; Helbing, 2001). From 2005 to 2006, more than fifty articles on various topics of traffic research had been published in the physics journal **Physica A** alone. With a keyword search in the journal **Physica A**, there were only 7 papers related to traffic dynamics in the publication of 2004. In 2005, the number increased to 17; the number further advanced to 36 in both 2006 and 2007. One of the research interests concerns the intrinsic fluctuations of the dynamics. For the highway system, the dynamics could be simple and deterministic; each vehicle is supposed to follow its preceder moving smoothly down the road. As vehicular density increases, however, a small fluctuation in one of the headways will lead to instability of the whole system. The congestion emerges inevitably without any specific causes, such as accidents or bottlenecks. One of the research interests in highway traffic is to provide a better understanding of the so–called phantom jam (Treiterer, 1975).

In contrast to the highway traffic, the city traffic is much more dynamic and complicated. Further considerations such as intersections and pedestrians should be included, e.g., traffic from different directions (Biham et al., 1992), operation of traffic lights (Huang & Huang, 2003b), and the interaction with pedestrians (Jiang et.al, 2006). It is interesting to note that the fluctuations can be nontrivial even in the case of a single vehicle (Nagatani, 2001b; Nagatani & Yoshimura, 2002; Nagatani, 2002c). The limited roadways are shared by more vehicles with all different itineraries. Within the city, vehicles are not expected to move smoothly down the road. At each intersection, pedestrians and vehicles from different directions would have to take turns to use the roadway or to yield to others. And the city layout, especially in the downtown area, is characterized by its full of intersections. Quite obviously, the bus transportation to replace numerous passenger cars and the traffic lights to regulate traffic at intersections are two vital ingredients to the city traffic.

We present a simple cellular automaton model to study the typical bus dynamics in a modern city. At the first stage, the nontrivial fluctuations are prescribed by the stochastic moving of bus interacted with the stochastic arrival of passengers. As passengers increase, the bus schedule shows a clear transition. Both numerical and analytical results are presented. The divergence of bus schedule can be taken as an analogy to the gridlock of 4-way traffic. We also comment on the strategy to keep a stable schedule.

At the second stage, we examine the bus schedule interrupted by the traffic lights. We analyse the city buses time headway distribution and compare to the real time headway measurements. Since experimental data shows neither a smooth nor a random distribution,

a mean–field theory with effects of traffic lights is developed to explain the peculiarity. It is shown that a smooth distribution can be modified significantly by the operation of traffic lights, and matches the experimental data characteristics well. It is therefore concluded that the posted average time-headway is of not much help for the passengers expecting the next bus at the bus stop.

2. Dynamics of a cyclic bus

First, we focus on an interesting case where a cycling bus moves along a closed route and interacts with the passengers waiting to get on the bus. We propose a cellular automaton model to study the fluctuations of the dynamics.

Traffic dynamics does not involve fundamental forces of nature. Rather, the emergent phenomena can be taken as the collective behaviors involving human decisions. The basic researches of traffic dynamics are not aimed to reveal the fundamental interactions behind the dynamics, but to have an effective theory to capture the essence of phenomena. As a result, the same phenomena might be described in different theories by different languages. It would be interesting to compare these different theories and see how they are complementary for each other. There are various types of models being proposed in recent years. Judging by intuitions, the optimal velocity models (Bando et al., 1995) and the cellular automaton models (Schreckenberg et al., 1995) are most easily understood. These models are built on the microscopic behavior of each individual vehicle, which is governed either by ordinary differential equations or by operational rules. To the other end, one can also have a macroscopic theory, where the individual vehicles become irrelevant. The most popular one use the analogy to hydrodynamics described by partial differential equations (Kerner & Konhäuser, 1993). The details of each trajectory are smeared to have a macroscopic density and velocity field. More recently, a new approach was proposed. The schedule of an urban bus is taken as the basic variable. The recurrence schedule is described as a piecewise nonlinear map (Nagatani, 2001b; Nagatani, 2002c). In statistical physics, the correspondence between microscopic and macroscopic descriptions is always fascinating. The relationship between optimal velocity model and hydrodynamic model had been reported (Berg et al., 2000). In this section, we will explore the correspondence between cellular automata and nonlinear map.

We study the dynamics of a recurrent bus by a cellular automata. The model will be introduced in the following. Both numerical and analytical results are presented. We compare the results with previous findings from nonlinear map. The abrupt divergence can be reproduced and understood. However, we find that the critical value was overestimated in the nonlinear map. We also comment on a misleading strategy to stabilize the bus schedule.

2.1 Bus route model

We investigate the dynamics of a cycling bus with a cellular automata. The traveling bus is taken as a particle hopping along a discrete lattice periodically. Consider a cyclic bus route consists of M stops; at each bus stop, the passengers arrive at a rate γ. As a bus hops along the route, the hopping rate p is strongly influenced by the number of passengers N waiting at each stop. When N increases, p decreases accordingly to prescribe a delayed bus. In the original bus route model (O'Loan et al., 2000), there was no such a dependence. Later, a linear dependence was considered (Nagatani, 2002b). Subsequently, a much stronger dependence

was proposed (Nagatani, 2002c). Here, we adopt a simple quadratic form,

$$p = \frac{1}{1 + aN^2} \quad . \tag{1}$$

As a naive scaling of $(a \cdot \gamma^2)$ is expected, we assume $a = 1$ without loss of generality. In the model, there are only two parameters M and γ. When there is no passenger to delay the hopping, $\gamma = 0$, the bus completes the route at a fixed schedule $\Delta T = M$. The schedule ΔT is understood as the recurrence time of a cyclic bus on the route. As γ increases, ΔT is expected to increase accordingly. When a bus is delayed, there would be more passengers waiting at the bus stop; and as more passengers are accumulated, the bus would be further delayed. Thus, an instability can be expected as γ increases. When γ is larger than the critical value, ΔT diverges, i.e., the bus would never complete the route. With naive thinking, the divergence seems to be unrealistic. One would argue that the bus shall always arrive if you wait long enough.

In the mean-field approximation, the stochasticity is suppressed. With a schedule of ΔT_i, the average number of passengers waiting at each bus stop is $(\gamma \cdot \Delta T_i)$ and the next recurrence bus would spend an average time $[1 + (\gamma \cdot \Delta T_i)^2]$ there. Thus we obtain the following nonlinear map of a single variable ΔT,

$$\Delta T_{i+1} = M \left[1 + (\gamma \cdot \Delta T_i)^2 \right] \quad , \tag{2}$$

where the subscripts of ΔT denote the recurrence index. In this mean-field theory, ΔT diverges as γ increases. The critical value can be obtained as

$$\gamma > \frac{1}{2M} \quad . \tag{3}$$

With a small γ, the stable bus schedule is as following

$$\Delta T = \frac{1 - \sqrt{1 - 4\gamma^2 M^2}}{2\gamma^2 M} \quad . \tag{4}$$

A scaling of $(M \cdot \gamma)$ is also observed. In the limit of $\gamma \to 0$, the above analytic formula reproduces the fixed point of $\Delta T = M$; in the other limit of $\gamma \to 1/(2M)$, ΔT approaches its maximum of $(2M)$. In between these two limits, ΔT increases smoothly with the increase of γ.

In the cellular automaton simulations, the averaged ΔT increases with the increase of γ much faster than the prediction of mean-field theory. Obviously, the maximum at $\Delta T = 2M$ cannot be confirmed by numerical simulations. The critical value appears to be much less than the mean-field prediction at $\gamma = 1/(2M)$, see Fig. 1. With the microscopic simulations, the fluctuations can be easily observed. The typical results are shown in Fig. 2. For each recurrence i, wide fluctuations of ΔT_i can be noticed. In some cases, ΔT_i diverges and the bus will be delayed indefinitely. In Fig. 3, we plot the probability for a bus to have a stable schedule. A transition in between $\gamma = 0.01$ and $\gamma = 0.015$ can be observed. With a smaller γ, the bus recurs stably; with a larger γ, the schedule diverges easily.

To look into more details, we plot the probability distributions of ΔT at various γ, see Fig. 4. With the above mean-filed results, we would naively expect a simple distribution prescribing

an increasing ΔT with an increase γ. In stead, we observe an interesting distribution still dominated by the fixed schedule of $\Delta T = M$. As γ increases, the probability to keep the schedule decreases exponentially. Only when the dominant peak at $\Delta T = M$ subsides, the secondary distribution at $\Delta T > M$ becomes obvious. For the broad distribution of the secondary structure, the mean of ΔT shifts toward larger values as γ increases.
The exponential decay at the fixed schedule $\Delta T = M$ can be argued as following. The probability to keep the schedule can be expressed as

$$\left[\sum_{N=0}^{M} C_N^M (1-\gamma)^{M-N}\gamma^N \cdot \frac{1}{1+N^2}\right]^M \sim \exp\left(-\frac{M^2}{2}\,\gamma\right) \ . \tag{5}$$

The binomial distribution represents the probability of N passengers waiting at a bus stop in a time span of M. With these passengers, the bus has a finite probability to keep the schedule as prescribed in Eq. (1); and the same factor for each bus stop results in the power M. For a large M, the above analytic expression can be well approximated by a single exponential distribution. The numerical simulations can be fairly reproduced, especially for small γ, see Fig. 5.
For the secondary structure, the same probability considerations can also be applied to the delay of bus. For example, in a schedule of $\Delta T = M + 1$, the bus must be delayed in one of the M bus stops. The following expression can be obtained,

$$\begin{aligned}&\left[\sum_{N=0}^{M+1} C_N^{M+1}(1-\gamma)^{M+1-N}\gamma^N \cdot \frac{1}{1+N^2}\right]^{M-1} \\ &\times M\left\{\sum_{N=1}^{M} C_N^M (1-\gamma)^{M-N}\gamma^N \cdot \left(1-\frac{1}{1+N^2}\right)\cdot\left[\frac{\gamma}{1+(N+1)^2}+\frac{1-\gamma}{1+N^2}\right]\right\} \ ,\end{aligned} \tag{6}$$

where the first summation represents the probability of no delay in $(M-1)$ bus stops, and the delay in one of the bus stop is prescribed by the second summation. The factors in the square brackets ensure that the delay is only for one time step, i.e., the bus moves forward in the next time step, whether new passenger arrives or not. In the former case, we have a probability γ and the passenger number increases to $(N+1)$; in the latter case, we have a probability $(1-\gamma)$ and the passenger number remains at N. The above expression gives a simple profile peaked at $\gamma = 0.004$, see Fig. 6. Similar combinational probabilities can be applied to the further delay of bus. The numerical simulations can be reproduced. For the further delayed schedule, the peak shifts toward a larger value of γ.

2.2 Discussions

In this section, we propose a cellular automaton model to study the dynamics of a cycling bus. The intrinsic fluctuations in the traffic dynamics are prescribed by the stochastic moving of bus (along the route) coupled with the stochastic arrival of passengers (at the bus stops). We observe the wide fluctuations of the recurrent schedule. When the passengers increase, the schedule diverges suddenly, i.e., the bus was delayed indefinitely and unexpectedly. In reality, such phenomena can be found in the situations where the number of passengers waiting to get on the bus far surpasses the capacity of transportation. Some passengers would eagerly try to push themselves through the bus door but cannot, and the bus will not be able to drive

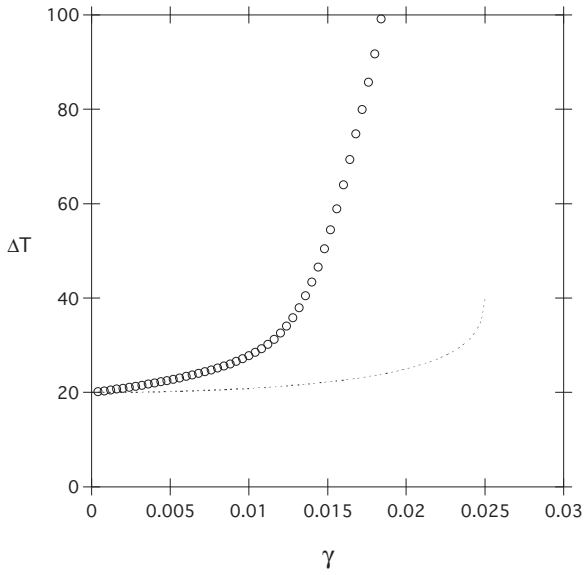


Fig. 1. Averaged recurrence schedule ΔT as a function of γ, where $M = 20$. The dotted line shows the mean-field prediction, which terminates at $\gamma = 1/(2M)$ with a maximum $\Delta T = 2M$.

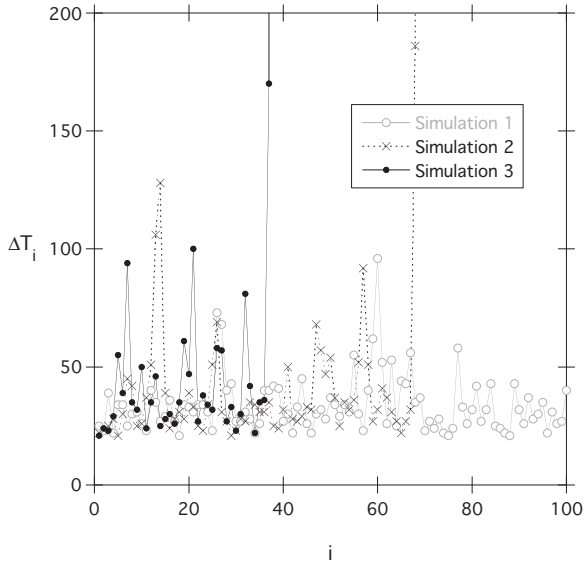


Fig. 2. Typical fluctuations of schedule ΔT_i at each recurrence i, where $M = 20$ and $\gamma = 0.013$. In the simulation 1, the schedule is stable; in the simulation 2, the schedule diverges at $i = 69$; in the simulation 3, the schedule diverges at $i = 39$.

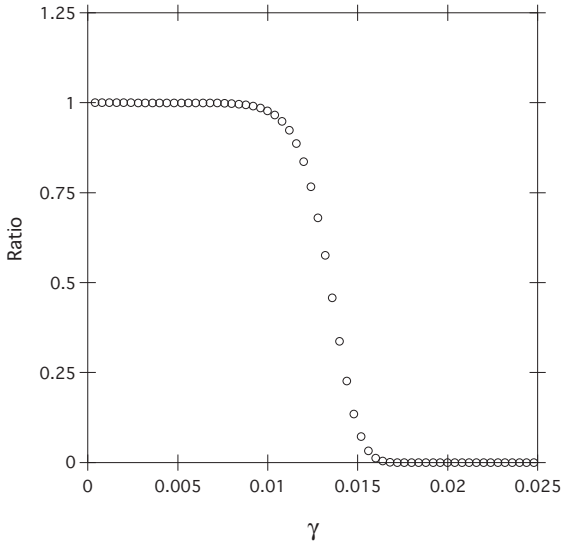


Fig. 3. Ratio of stable schedule as a function of γ, where $M = 20$. A stable schedule is defined as being able to recur at $i = 100$ (as simulation 1 in Fig. 2), where the divergence is taken as $\Delta T_i > 10M$ (as simulations 2 and 3 in Fig. 2).

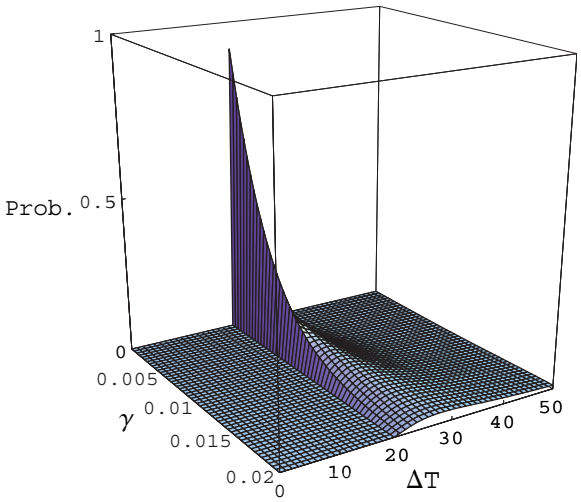


Fig. 4. Probability distributions of ΔT at various γ, where $M = 20$. A dominant peak at $\Delta T = M$ superimposes on the secondary structure at $\Delta T > M$.

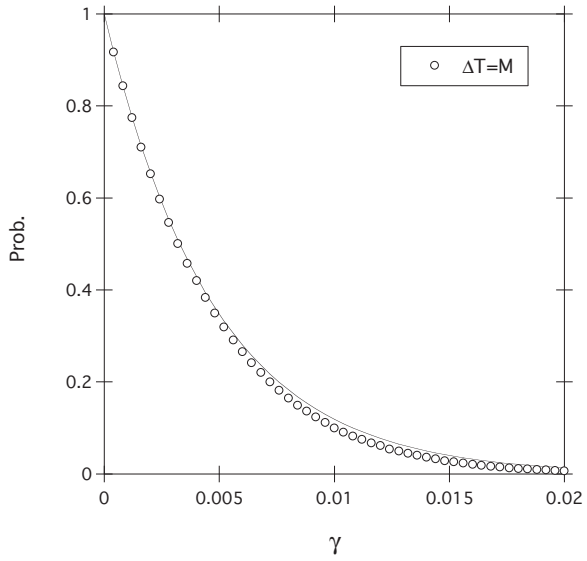


Fig. 5. Decay of the primary peak at $\Delta T = M$. The solid line shows the analytical result of a simple exponential distribution, Eq. (5).

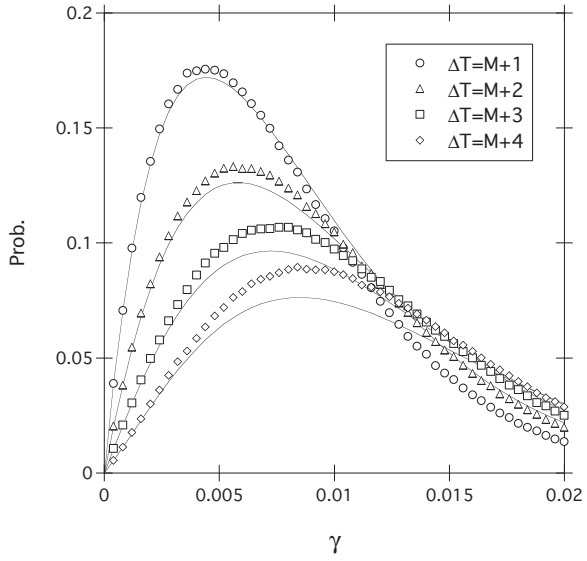


Fig. 6. Profile of the secondary bump at $\Delta T > M$. The solid lines show the analytical results from combinational probabilities.

away without properly closing its door. As a result, the indefinite delay emerges inevitably. The situation can be analogy to the gridlock appeared in an intersection where a few vehicles from different directions block each other and all the traffic is stopped indefinitely.
We also compare the results with previous finding based on the nonlinear maps, which prescribes an abrupt transition at $\gamma = 1/(2M)$. In the cellular automaton simulations, where the fluctuations were properly taken care of, we observe a smoother transition at a much small critical value. In the deterministic mean-field theory, the critical value was overestimated by a factor of two. As the fluctuations play an important role in traffic phenomena, the conjectures based on deterministic theory can be misleading sometimes. For example, a strategy was proposed to avoid the divergent schedule: by skipping a few stops, the bus will be able to keep a stable schedule (Nagatani, 2002c). As shown in this study, the stable schedule can only be reached by limiting the value $(M \cdot \gamma)$. With a fixed γ, i.e., fixed passenger arrival rate, the only option is to reduce the number of stops M. We point out that the conclusion in the above Reference was based on a unrealistic presumption: when the bus skips a stop, those passengers waiting at the bus stop disappear. Thus such an effectively reducing M presumes an effectively reducing γ. The problem remains unsolved, unless we assume that those passengers would all leave the bus stop disappointedly to find other means of transportation whenever the bus skips the stop. Otherwise, there would be more passengers accumulated when the bus recurs later. We find that at a fixed γ, the only feasible strategy to stabilize the recurrent schedule is to add more buses to the route. It is well known that the same instability will lead these buses to bunch together as the passengers increase (O'Loan et al., 1998; Nagatani, 2002a). By instructing the bus drivers to skip a few stops will now keep these buses more or less equal distanced, which would provide an effectively reduced M without presuming a reduced γ. Thus the stable scheme can be restored by the strategy of adding more buses adjoined with skipping a few stops when necessary.

3. Time headway distribution of city buses

Public transportation and traffic signal are two important issues of city traffic. In most previous studies, these two issues were often addressed separately. In references (Brockfeld et al., 2001; Huang & Huang, 2003a; Huang & Huang, 2003b; Tan et al., 2004; Toledo et al., 2004; Jiang & Wu, 2005; Nagatani, 2005a; Nagatani, 2005b; Nagatani, 2005c; Jiang & Wu, 2006; Nagatani, 2006a; Nagatani, 2006b; Toledo et al., 2007; Nagatani, 2007a; Nagatani, 2007b; Nagatani, 2007d; Nagatani, 2007e; Nagatani, 2008), the impacts of traffic lights have been studied in some details. Yet the main concern is on the passenger cars, not the public transportation. In references (O'Loan et al., 1998; Desai & Chowdhury, 2000; Nagatani, 2000; Nagatani, 2001a; Nagatani, 2001c; Huijberts, 2002; Nagatani, 2002d; Hill, 2003; Nagatani, 2003a; Nagatani, 2003b; Nagatani, 2003c; Nagatani, 2003d; Nagatani, 2003e; Nagatani, 2006c; Yuan et al., 2007; Nagatani, 2007c), various models for bus transportation have been proposed. Again, most of the research focus on the interactions between bus and passengers, with the operation of traffic lights neglected.
In this section, we address these two issues in a framework. We study the bus dynamics influenced by the operation of traffic lights. The public transportation and the passenger vehicles are distinctly different in dynamics. Basically, a passenger car would prefer to have a non–stop journey from its origin to destination, while a bus has to stop at every bus stop to load and unload passengers. To provide a reliable service of public transportation, keeping

schedule is all important. On every bus stop, the most needed information should be the schedule. From our daily experiences, however, the buses never seem to keep the schedule. For a bus stop posting a 5–minute regular schedule, all too often you would have to wait much longer than 5 minutes for a bus to arrive. And when the buses do arrive, quite often they come in abundance, i.e., arriving one after another. For a passenger expecting a bus every 5 minutes, one might wonder why the buses like to bunch all together. And if it weren't a strict 5 minutes interval, one might reason that it should at least be a Gaussian–like distribution centered at 5 minutes. But the experience seems to indicate that neither a smooth distribution nor a purely random distribution is the case. In the following, we will attempt to answer this question. We will explore the intrinsic instability of bus schedule and the statistical meaning of this 5–minute interval. Both theoretical analysis and experimental data are presented. Although the traffic phenomena can be observed everywhere, concrete data are limited in the literature. Even the experimental data for the above mentioned phantom jam were only reported very recently (Sugiyama et al., 2008). To our knowledge, we are only aware of the data from Mexican buses (Krbálek & Šeba, 2000; Krbálek & Šeba, 2003). In the following, we present a mean–field theory for the bus schedule interfered by the traffic lights. The experimental data and theoretical analysis are also presented. We reinterpret the data in references (Krbálek & Šeba, 2000; Krbálek & Šeba, 2003) to demonstrate the validity of our theory.

3.1 Mean–field theory

In an ideal world, the buses should keep a perfect schedule. The distribution of time headway can be represented by a Dirac delta function as

$$P(t) = \delta(t - T_0) \ , \tag{7}$$

where T_0 denotes the regular schedule or the time headway, which is understood as the interval between two consecutive buses passing the same location. On the other hand, in a chaotic world, buses are not aware of their mutual positions, i.e., there is no correlation, and the distribution of time headway becomes Poissonian as

$$P(t) = \frac{1}{T_0} \exp\left(-\frac{t}{T_0}\right) \ . \tag{8}$$

It describes a high possibility of buses bunching together, because the leading bus tends to take more time to collect more passengers and the following bus gradually catches up. In between these two limits, the Wigner distribution has been proposed to describe the time headway as (Krbálek & Šeba, 2000; Krbálek & Šeba, 2003)

$$P(t) = \frac{32}{\pi^2 T_0^3} t^2 \exp\left(-\frac{4t^2}{\pi T_0^2}\right) \ , \tag{9}$$

which is normalized to give an expectation value of T_0. This is the case that there is correlation between buses, and their mutual awareness keep them away from bunching together. The three typical distributions are shown in Fig. 7.

Now we consider the influence from the operation of a traffic light, which is simply taken as a signal switching periodically between green and red. We denote the duration of green phase and red phase respectively as T_G and T_R. Basically the traffic light cycle and the bus schedule are of the same order of magnitude, i.e., a few minutes. If it arrived at the intersection in the

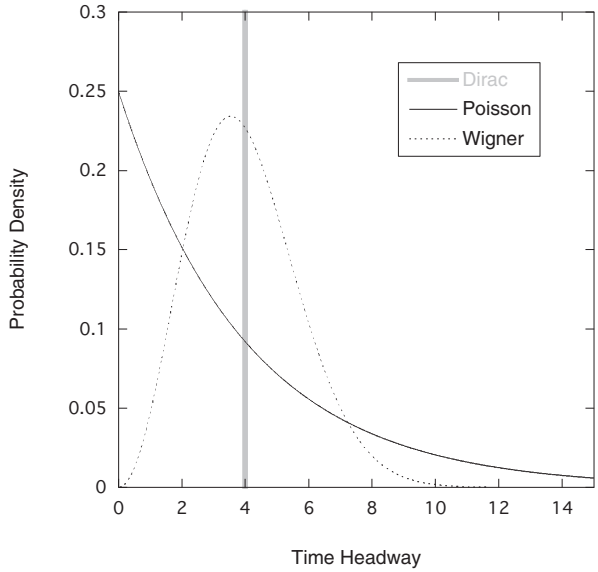


Fig. 7. Three typical time headway distributions: Dirac delta distribution for a perfect schedule, Poisson distribution for a chaotic schedule, and Wigner distribution in between these two limits.

green phase, the bus will move forward uninterruptedly. If it arrived in the red phase, the bus will have to stop at the intersection waiting for the signal to turn green. Accordingly the time headway to the preceding bus is lengthened and the time headway to the following bus is shortened. In such ways, the operation of a traffic light can change the headway distribution significantly. In the following, we derive this changed distribution $P'(t)$ in a mean–field theory. The probability for two buses to bunch together, i.e., a zero time headway, can be written as

$$P'(0) = \frac{1}{(T_G + T_R)} \left[\int_0^{T_G} dt_1 \, P(0) + \int_0^{T_R} dt_1 \int_{t_1}^{T_R} dt_2 \, P(t_2 - t_1) \right] , \tag{10}$$

where the first and second terms represent the contributions from a bus arriving in the green and red phases, respectively. In the green phase, buses are uninterrupted. The probability of bunching together can be retrospected to $P(0)$, which prescribes the probability of bunching together when the traffic light is absent. In the red phase, the bunching probability can be further enhanced. Two buses are forced to bunch together when they both arrive in the same red phase. In the second term of above formulation, the arriving times of these two buses are denoted by t_1 and t_2. The two integrations with the same upper limit at T_R ensure that the two buses arrive in the same red phase. Obviously, the buses in an ideal world will not bunch together when the red phase is shorter than the schedule, i.e., $T_R < T_0$. With a Wigner distribution, the buses begin to bunch together noticeably by the operation of a traffic light,

i.e., $P'(0) > 0$ while $P(0) = 0$. In the chaotic world prescribed by the Poisson distribution, the bunching probability is significantly enhanced by the traffic lights.
By similar approaches, the distribution for a small time headway t can be written as

$$P'(t) = \frac{1}{(T_G + T_R)} \left[\int_0^{T_G - t} dt_1 \, P(t) + \int_0^{T_R} dt_1 \, P(t + T_R - t_1) \right] , \tag{11}$$

where again the first and the second terms represent the contributions from buses arriving in the green and the red phases respectively. In the green phase, a bus arrives at t_1; the following bus is separated by an interval t. The two buses will have no trouble keeping the time headway t if they both arrive in the same green phase. The integration has an upper limit at $t_1 = (T_G - t)$ to ensure that the following bus will not be interrupted by the traffic light. In the red phase, a bus arriving at t_1 will be delayed by $(T_R - t_1)$ until the traffic light switches from red to green. Thus a following bus separated by an internal $(t + T_R - t_1)$ originally will be catching up and has a time headway t finally. We note that the limit $t \to 0$ in Eq. (11) does not reproduce the result of Eq. (10). Extra contributions from two buses arriving within the same red phase are taken into account by the double integration in Eq. (10). While to maintain a finite time headway in Eq. (11), the following bus must arrive in the green phase. In the limit $t \to 0$, Eq. (11) only takes into account the contribution from a following bus arriving right at the switching of traffic light.
The validity of Eq. (11) is limited by $t < T_G$, which is obvious from the upper limit of the first integration. If the time headway is larger than T_G, these two buses cannot pass the intersection within the same green phase. When the traffic light has a long red phase, some of the time headways cannot be observed. Specifically, in the cases of $T_G < T_R$, we have $P(t) = 0$ for $T_G < t < T_R$. With the operation of traffic lights, a smooth distribution $P(t)$ will be changed to a discontinuous one $P'(t)$. As the time headway t further increases, the distribution $P'(t)$ is shaped by different mechanisms. For $t > T_R$, the distribution can be written as

$$P'(t) = \frac{1}{(T_G + T_R)} \left[\int_{T_G + T_R - t}^{T_G} dt_1 \, P(t) + \int_0^{T_R} dt_1 \, P(t - T_R + t_1) \right] , \tag{12}$$

where both terms are for a bus arriving in the green phase; the first and the second integrations take into account the contributions when the following bus arrives in the green phase and in the red phase respectively. As can be seen, in the first integration, the distribution $P(t)$ has not been modified. With a bus arriving at t_1 in the green phase, the following bus separated by a time headway t will pass the same intersection smoothly in the next green phase. Obviously, t_1 is lower bounded. If the bus arrived too early in the green phase, a separation of interval t will result in the following bus arriving in the red phase. The validity of Eq. (12) is then set by the lower limit of this integration, i.e., $t < (T_G + T_R)$. In the second integration, the distribution $P(t)$ has been modified. A time headway t is the result when a bus arrives at $(T_G + T_R - t)$ in the green phase and the following bus arrives at t_1 in the subsequent red phase. As the following bus is delayed by $(T_R - t_1)$, the time headway is lengthened from $(t - T_R + t_1)$ to t. To have a time headway within the range $T_R < t < (T_G + T_R)$, the bus cannot arrive in the

red phase. If a bus arrived in the red phase and the following bus arrived in the subsequent green phase, the time headway is too short, as prescribed in Eq. (11), while if the following bus arrived in the next red phase, the time headway will become $(T_G + T_R)$, which is the limit of applying Eq. (12).

As the traffic light switches periodically, the modification of $P(t)$ also goes through the same cycle. At the time headway $(T_G + T_R)$, the distribution can be written as

$$P'(T_G + T_R) = \frac{1}{(T_G + T_R)} \left[\int_0^{T_G} dt_1 \, P(T_G + T_R) + \int_0^{T_R} dt_1 \int_0^{T_R} dt_2 \, P(T_G + T_R + t_2 - t_1) \right] , \tag{13}$$

which is basically the extension of Eq. (10) to the next cycle. In the first term, a bus arrives in the green phase and the following bus arrives in the next green phase. In the second term, a bus arrives in the red phase and the following bus arrives in the next red phase. The distribution for an even larger time headway can be obtained similarly.

The typical results for the modified Wigner distribution are shown in Fig. 8. The

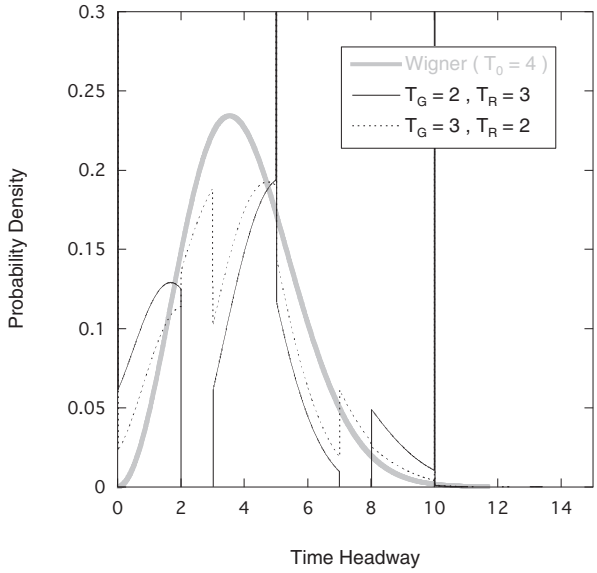


Fig. 8. Interference of a Wigner distribution by the traffic light.

discontinuities at $t = T_G$ and $t = T_R$ are obvious. In the cases of $T_G < T_R$, the distribution is depleted in the range $T_G < t < T_R$; in the opposite cases of $T_G > T_R$, the distribution is enhanced for $T_R < t < T_G$. The spikes at a multitude of $t = (T_G + T_R)$ are the results of two consecutive buses being both stopped by the red light. This can be taken as a kind of condensation where a small adjustment in the arriving time will make no difference. Similar modifications to the Poisson distribution are shown in Fig. 9. In the following, we will present some experimental data to support these peculiar distributions.

3.2 Experiment and analysis

In addition to three lines of Metro Rapid Transit (MRT) rail system, there are more than 400 city bus lines in the Taipei metropolitan area, serving 6 million people living in Taipei County and Taipei City, the capital of Taiwan. Most buses operate from early morning to late night, an unusually long day among major cities of the world. This provides an excellent opportunity to observe and analyse all the traffic phenomena. To better observe the effect of traffic lights on time headway distribution, we pick a busy bus line with short schedule interval in the order of traffic light period. The observation was carried out near a typical bus stop on the route 307 of Taipei City Bus. As there are designated bus lanes in most major avenues of Taipei, see Fig. 10, the interactions between buses and passenger vehicles are minimized. As one of the most popular buses, 307 runs between Taipei County and Taipei City separated by Tamsui River, transporting people living on both sides of the river. The whole route takes between 3 and 4 hours to complete, with more than 50 stops along the way. The observation point is about one third from the starting stop on Taipei City side. The departure time for each bus was recorded; the arrival time is unregistered. The data are then converted into the so–called time headway, i.e., the time interval between the departure of two consecutive buses. There are 3210 data points collected during December 2005. Partly owing to the popularity of this bus route, only a slight difference in the weekday rush hours is observed, and all the data are included in the following analysis. The distribution of time headway is shown in Fig. 11.

With an average time headway of 4 minutes, the longest interval observed is 25 minutes. The distribution is far from a single peak centered at the averaged time headway. Nor does the distribution resemble a monotonically decreasing curve, which has been proposed by considering the interactions between consecutive buses. The oscillatory behavior is obvious. Such oscillatory patterns can be attributed to the interference from a nearby traffic light. When the traffic light turns red, buses are stopped and accumulated, resulting in the prominent peak

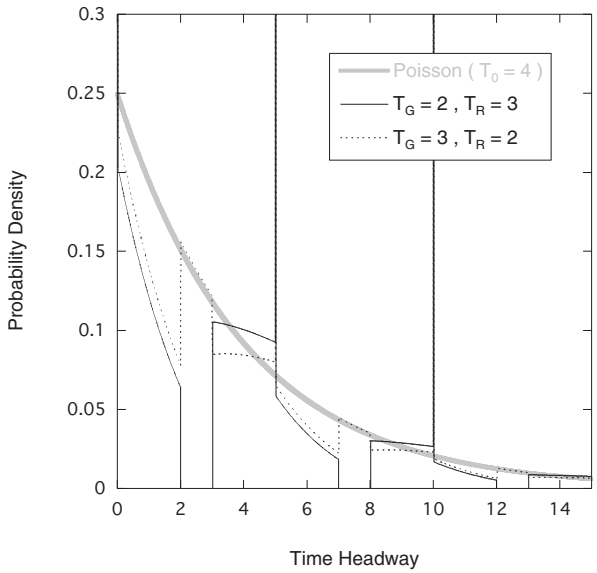


Fig. 9. Interference of a Poisson distribution by the traffic light.

Fig. 10. Typical Taipei City bus lanes.

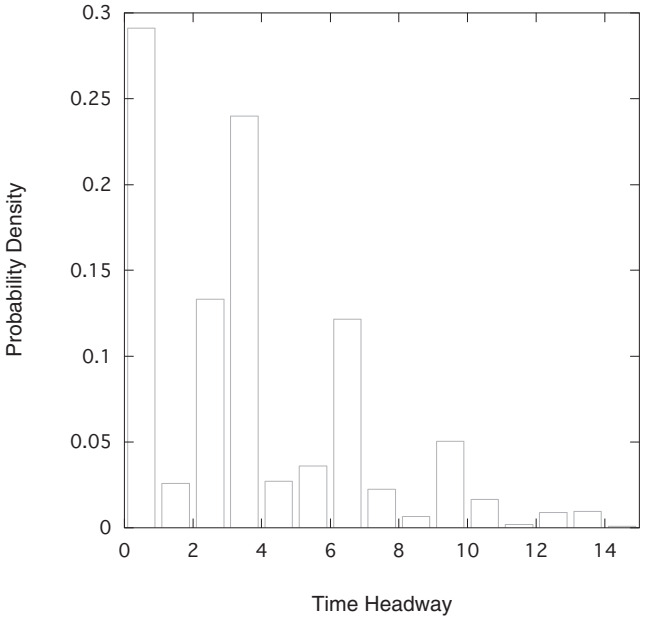


Fig. 11. Typical time headway distribution for the route 307 of Taipei City Bus.

near the zero time headway. As the traffic light switches between green and red alternatively, it acts as a periodic chopper. Certain intervals will not be able to pass through the chopper easily, which results in the several dips in the distribution. A quantitative description from the simple formulation of the last section is shown in Fig. 12.

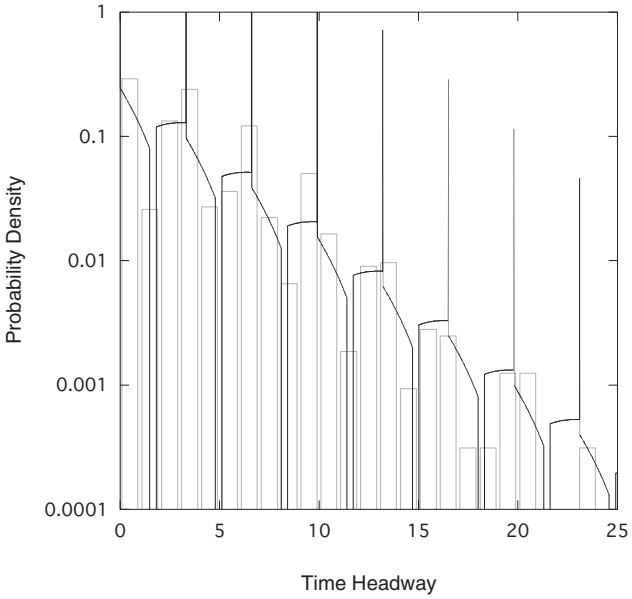


Fig. 12. The data of Fig. 11 described by a modified Poisson distribution on a semi-logarithmic scale. The parameters are $T_G = 1.5$, $T_R = 1.8$ and $T_0 = 3.6$ min.

With the number of data points, the fluctuations are averaged out, and the distribution represents true behavior, as can be seen more clearly on a semi-logarithmic scale. We do not expect such a naive mean–field approximation to provide an accurate description to the empirical data, yet the characteristic features are well captured. The narrow spikes are the artifact of a continuous theory. With more refined treatment, multiple peaks with a finite width can be expected.

We find that such peculiar peaks and dips can also be discerned from the existing data. In references (Krbálek & Šeba, 2000; Krbálek & Šeba, 2003), the time headway distributions are described by smooth curves, a Poisson distribution and a Wigner distribution respectively. The deviations from the smooth curves might be attributed to random fluctuations. On the other hand, it is interesting to note that such deviations can be produced by taking into account of the influences of traffic lights. The results are shown in Figs. 13 and 14. We do not attempt to provide a better fitting. The point is just to emphasize that the drastic highs and lows in the distribution can be attributed to the influences of traffic lights.

3.3 Conclusion

In this section, we study the bus schedule interfered by the operation of traffic lights. The city traffic is often contrasted with the highway traffic. However, they both share a characteristic feature of intrinsic instability. The highway congestion may emerge out of nowhere. Similarly

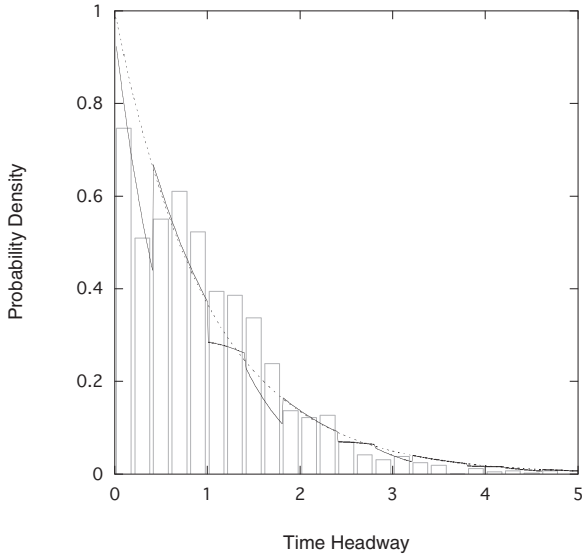


Fig. 13. The data of the buses in Puebla (Krbálek & Šeba, 2000; Krbálek & Šeba, 2003). The dotted line shows the Poisson distribution. The parameters are $T_G = 1$, $T_R = 0.4$ and $T_0 = 1$ for the solid line.

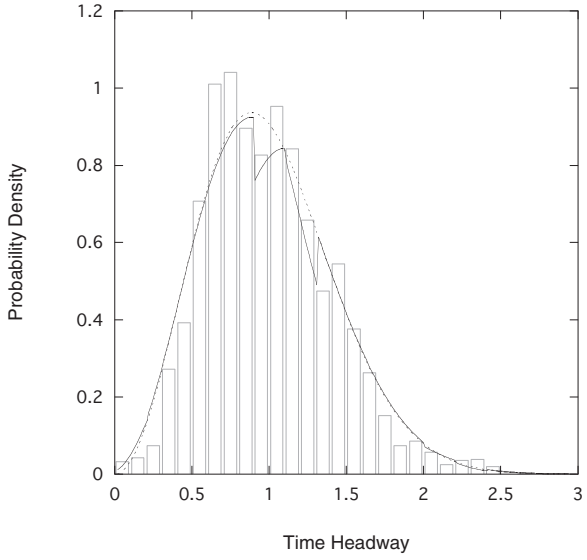


Fig. 14. The data of the buses in Cuernavaca (Krbálek & Šeba, 2000; Krbálek & Šeba, 2003). The dotted line shows the Wigner distribution. The parameters are $T_G = 0.9$, $T_R = 0.2$ and $T_0 = 1$ for the solid line.

the city buses may bunch together without obvious cause. The bus schedule can be influenced by all sorts of possibilities. With a simple mean–filed approximation and preliminary measurements, we show that the time headway distribution can be modified significantly by a nearby traffic light. For most public transportation, the buses on a given route are dispatched with a fixed time headway, and this fixed time headway is often provided at every bus stop as the most important information to passengers. However, as buses maneuver around the city, the headways will be modified drastically. The local influences, especially from a nearby traffic light, can be prominent. Thus the headway distribution can be very different for different stop. With this observation, we conclude that the posted time headway conveys little information to the passengers waiting by the bus stop. In most Taipei City bus stops, an intelligent transportation system has been implemented, displaying the location of each bus on the route and the estimated time of arrival. For providing a more useful and reliable information to the passengers, this is the way to go.

4. References

Bando, M., Hasebe, K., Nakayama, A., & Shibata, A., & Sugiyama, Y. (1995) *Phys. Rev.* E51, pp.1035.

Berg, P., Mason, A., & Woods, A. (2000) *Phys. Rev.* E61, pp.1056.

Biham, O., Middleton, A. A., & Levine, D. (1992) *Phys. Rev.* A46, pp.R6124.

Brockfeld, E., Barlovic, R., Schadschneider, A., & Schreckenberg, M. (2001) *Phys. Rev.* E64, pp.056132.

Chowdhury, D., Santen, L. & Schadschneider, A. (2000) *Phys. Rep.* 329, pp.199.

Desai, R. C. & Chowdhury, D. (2000) *Eur. Phys. J.* B15, pp.375.

Helbing, D. (2001) *Rev. Mod. Phys.* 73, pp.1067.

Hill, S. A. (2003) *Physica* A328, pp.261.

Huijberts, H. J. C. (2002) *Physica* A308, pp.489.

Huang, D. W. & Huang, W. N. (2003a) *Int. J. Mod. Phys.* C14, pp.539.

Huang, D. W. & Huang, W. N. (2003b) *Phys. Rev.* E67, pp.056124.

Jiang, R., Helbing, D., Shukla, P. K., & Wu, Q. S. (2006) *Physica* A368, pp.568.

Jiang, R. & Wu, Q. S. (2005) *Physica* A355, pp.551.

Jiang, R. & Wu, Q. S. (2006) *Physica* A364, pp.493.

Kerner, B. S. & Konhäuser, P. (1993) *Phys. Rev.* E48, pp.R2335.

Krbálek, M. & Šeba, P. (2000) *J. Phys. A* 33, pp.L229.

Krbálek, M. & Šeba, P. (2003) *J. Phys. A* 36, pp.L7.

Maerivoet, S. & De Moor, B. (2005) *Phys. Rep.* 419, pp.1.

Nagatani, T. (2000) *Physica* A287, pp.302.

Nagatani, T. (2001a) *Physica* A296, pp.320.

Nagatani, T. (2001b) *Physica* A297, pp.260.

Nagatani, T. (2001c) *Phys. Rev.* E63, pp.036115.

Nagatani, T. (2002a) *Physica* A305, pp.629.

Nagatani, T. (2002b) *Physica* A312, pp.251.

Nagatani, T. (2002c) *Physica* A316, pp.637.

Nagatani, T. (2002d) *Phys. Rev.* E66, pp.046103.

Nagatani, T. (2003a) *Physica* A321, pp.641.

Nagatani, T. (2003b) *Physica* A322, pp.685.

Nagatani, T. (2003c) *Physica* A323, pp.686.
Nagatani, T. (2003d) *Physica* A327, pp.570.
Nagatani, T. (2003e) *Phys. Rev.* E68, pp.036107.
Nagatani, T. (2005a) *Physica* A347, pp.673.
Nagatani, T. (2005b) *Physica* A348, pp.561.
Nagatani, T. (2005c) *Physica* A350, pp.577.
Nagatani, T. (2006a) *Physica* A361, pp.619.
Nagatani, T. (2006b) *Physica* A368, pp.560.
Nagatani, T. (2006c) *Physica* A371, pp.683.
Nagatani, T. (2007a) *Physica* A374, pp.419.
Nagatani, T. (2007b) *Physica* A377, pp.651.
Nagatani, T. (2007c) *Physica* A377, pp.661.
Nagatani, T. (2007d) *Physica* A380, pp.503.
Nagatani, T. (2007e) *Physica* A386, pp.381.
Nagatani, T. (2008) *Physica* A387, pp.1637.
Nagatani, T. & Yoshimura, J. (2002) *Physica* A316, pp.629.
O'Loan, O. J., Evans, M. R., & Cates, M. E. (1998) *Phys. Rev.* E58, pp.1404.
Schreckenberg, M., Schadschneider, A., Nagel, K., & Ito, N. (1995) *Phys. Rev.* E51, pp.2939.
Sugiyama, Y., Fukui, M., Kikuchi, M., Hasebe, K., Nakayama, A., Nishinari, K., Tadaki, S., & Yukawa, S. (2008) *New. J. Phys.* **10**, 033001.
Tan, H. L., Zhang, C. Y., Kong, L. J., & and Liu, M. R. (2004) *Int. J. Mod. Phys.* B18, pp.2658.
Toledo, B. A., Munoz, V., Rogan, J., Tenreiro, C., & Valdivia, J. A. (2004) *Phys. Rev.* E70, pp.016107.
Toledo, B. A., Cerda, E., Rogan, J., Munoz, V., Tenreiro, C., Zarama, R., & Valdivia, J. A. (2007) *Phys. Rev.* E75, pp.026108.
Treiterer, J. (1975) *Ohio State Technical Report* No. PB 246 094.
Yuan, Y. M., Juan, R., Wu, Q. S., & Wang, R. L. (2007) *Int. J. Mod. Phys.* C18, pp.1925.

11

Application of Cellular Automaton Model to Advanced Information Feedback in Intelligent Transportation Systems

Chuanfei Dong[1] and Binghong Wang[2,3]
[1]*Georgia Institute of Technology*
[2]*University of Science and Technology of China*
[3]*University of Shanghai for Science and Technology and Shanghai Academy of System Science*
[1]*U.S.A.*
[2,3]*P.R.China*

1. Introduction

For some socioeconomic systems, it is desirable to provide real-time information or even a short-term forecast about dynamics. For instance, in stock markets it is advantageous to give a reliable forecast in order to maximize profit. In traffic flow, advanced traveler information systems (ATIS) provide real-time information about the traffic conditions to road users by means of communication such as variable message signs, radio broadcasts, or on-board computers (Adler & Blue, 1998). The aim is to help individual road users to minimize their personal travel time. Therefore traffic congestion should be alleviated, and the capacity of the existing infrastructure could be used more efficiently. Fig. 1 shows a schematic diagram of an information feedback system, which demonstrates that feedback information plays a significant role in the loop.

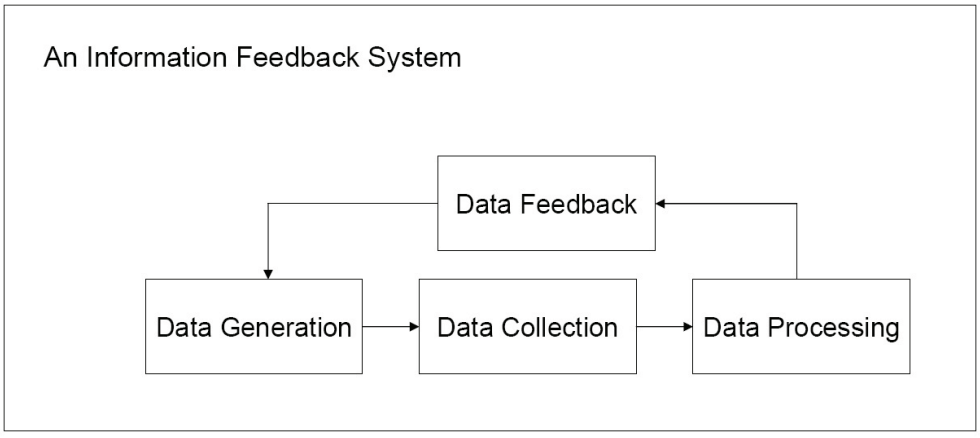


Fig. 1. The schematic diagram of an information feedback system.

Physics, other sciences and technologies meet at the frontier area of interdisciplinary research (Helbing, 1996; Chowdhury et al., 2000; Helbing, 2001; Nagatani, 2002). The concepts and techniques of physics are being applied to such complex systems as traffic systems. A lot of theories have been proposed such as car-following theory (Rothery, 1992), kinetic theory (Prigogine & Andrews, 1960; Paveri-Fontana, 1975; Helbing & Treiber, 1998) and particle-hopping theory (Nagel & Schreckenberg, 1992; Biham et al., 1992; Blue & Adler, 2001). These theories provide insights that help traffic engineers and other professionals to better manage congestion. Therefore these theories indirectly make contributions to alleviating traffic congestion and enhancing the capacity of existing infrastructure. Although dynamics of traffic flow with real-time traffic information have been extensively investigated (Friesz et al., 1989; Arnott et al., 1991; Ben-Akiva et al., 1991; Mahmassani & Jayakrishnan, 1991; Kachroo & Özbay, 1996; Yokoya, 2004), finding out a more efficient feedback strategy is still an overall task. Recently, some information feedbacks have been proposed to investigate the two-route scenario with the same length. Wahle *et al.* (2000 & 2002) firstly investigated the two-route scenario with travel time feedback strategy (TTFS). Subsequently, Lee *et al.* (2001) studied the effect of a different type of information feedback (MVFS), i.e. instantaneous average velocity. Wang *et al.* (2005) proposed a third type of information feedback (CCFS), i.e. instantaneous congestion coefficient which is defined as

$$C = \sum_{i=1}^{q} n_i^2. \tag{1}$$

where, n_i stands for vehicle number of the ith congestion cluster in which cars are close to each other without a gap between any two of them; q is the number of congestion clusters on the route. Then Dong *et al.* (2010b) put forward another type of information feedback (WCCFS), i.e. instantaneous weighted congestion coefficient which is defined as

$$C_w = \sum_{i=1}^{p} F(n_m) n_i^2. \tag{2}$$

where the definition of n_i is the same as above, $F(n_m)$ is the weight function, andn_m stands for the position of the ith congestion cluster. Here, we use the result of median rounding $\lfloor n_m \rfloor$ of the ith congestion cluster to represent its position. Furthermore, in order to provide road users with better guidance, Dong *et al.* (2009a; 2009b; 2010a; 2010d) proposed another two types of information feedback strategies named corresponding angle feedback strategy (CAFS) and prediction feedback strategy (PFS), respectively. The corresponding angle coefficient is defined as

$$C_\theta = \sum_{i=1}^{q} \theta_i^2 = \sum_{i=1}^{q} \left(\arctan(\frac{n_i^{first}}{H}) - \arctan(\frac{n_i^{first} - l_i}{H}) \right)^2. \tag{3}$$

where n_i^{first} stands for the position of the first vehicle in the ith congestion cluster, in which vehicles are close to each other without a gap between any two of them. l_i and θ_i denote the length and the weight (corresponding angle) of the ith congestion cluster, respectively. H denotes the vertical distance from point T to the route (see Fig.2). In this chapter, we set $H = 100$. PFS is based on CCFS, and the predicted congestion coefficient (C_p) is defined as:

$$C_p(t) = C(t + \Delta t) = \sum_{i=1}^{q} n_i^2(t + \Delta t). \tag{4}$$

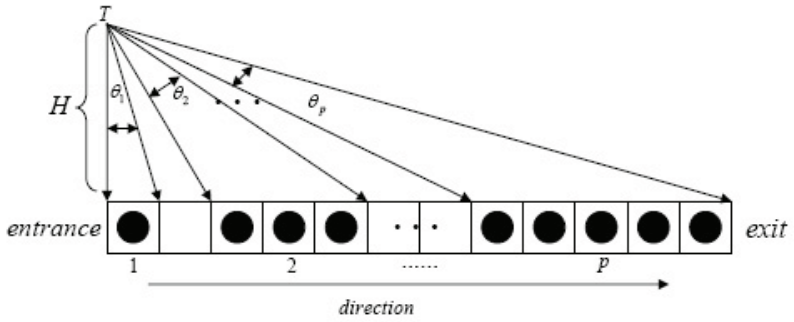


Fig. 2. Angles corresponding to different congestion clusters on the lane.

which indicates that PFS uses future road condition (the value of congestion coefficient C at time $t + \Delta t$) as feedback information.

It has been proved that TTFS is the worst one which brings a lag effect to make it impossible to provide the road users with the real situation of each route (Lee et al., 2001); CCFS is more efficient than MVFS because the random brake mechanism of the Nagel-Schreckenberg (NS) model (Nagel & Schreckenberg, 1992) brings fragile stability of velocity (Wang et al., 2005); WCCFS is more efficient than CCFS for the reason that CCFS does not take the weights of different parts of the route into consideration (Dong et al., 2010b). However, WCCFS is still not the best one due to the fact that the weight function $F(n_m)$ does not contain any information related to the length of the congestion cluster while the corresponding angle θ takes both the length and route location of each cluster into account (Dong & Ma, 2010a). Though the road capacity adopting PFS is the best one among these feedback strategies (Dong et al., 2009a; Dong et al., 2009b; Dong et al., 2010d), the validity of PFS depends on the length of the route when the traffic system is multi-route (Dong et al., 2010d). Also, PFS is not easy to realize since it needs predicted road data, which will be discussed in detail in the following paragraphs. We report the simulation results adopting six different feedback strategies TTFS, MVFS, CCFS, WCCFS, CAFS and PFS in a two-route scenario with a single route following the NS mechanism.

The chapter is arranged as follows: In Section 2, several cellular automaton models of traffic flow will be mentioned including some analytical studies and a two-route scenario is briefly introduced, together with six feedback strategies of TTFS, MVFS, CCFS, WCCFS, CAFS and PFS all depicted in detail. In Section 3, some simulation results will be presented and discussed based on the comparison of six different feedback strategies. In Section 4, we will make some conclusions.

2. The model and feedback strategies

A. NS mechanism

Recently, a lot of cellular automaton models are proposed such as NS model (Nagel & Schreckenberg, 1992), FI model (Fukui & Ishibashi, 1996), VDR model (Barlovic et al., 1998), VE model (Li et al., 2001), BL model (Knospe et al., 2000), VE model (Li et al., 2001), Kerner-Klenov-Wolf model (Kerner et al., 2002; Kerner & Klenov, 2004; Kerner, 2004), FMCD model (Jiang & Wu, 2003; Jiang & Wu, 2005), VDDR model (Hu et al., 2007), and VA model (Gao et al., 2007). Also, a lot of analytical works based on statistical physics, such as the

spacing-oriented mean field theory, have been studied to investigate fundamental diagrams and asymptotic behavior of CA models, i.e., NS model, FI model, and NS & FI combined CA model (Wang et al., 2000a; Wang et al., 2000b; Mao et al., 2003; Wang et al., 2003; Wang et al., 2001; Fu et al., 2007). Among these CA models, the NS model is so far the most popular and simplest cellular automaton model in analyzing the traffic flow (Nagel & Schreckenberg, 1992; Chowdhury et al., 2000; Helbing, 2001; Nagatani, 2002; Wang et al., 2002), where the one-dimension CA with periodic boundary conditions is used to investigate highway and urban traffic. This model can reproduce the basic features of real traffic like stop-and-go wave, phantom jams, and the phase transition on a fundamental diagram that plots vehicle flow versus density. Thus we still adopt NS model when comparing the effects of different feedback strategies in this chapter. In the following paragraphs, the NS mechanism will be briefly introduced as a basis of analysis.

The road is subdivided into cells (sites) with a length of Δx=7.5 m. The route length is set to be $L = 2000$ cells (corresponding to 15 km). N denotes the total number of vehicles on a single route of length L. The vehicle density can be defined as ρ=N/L. A time step corresponds to $\Delta t = 1s$, the typical time a driver needs to react. g_n(t) refers to the number of empty sites in front of the nth vehicle at time t, and v_n(t) denotes the speed of the nth vehicle, i.e., the number of sites that the nth vehicle moves during the time step t. In the present paper, we set the maximum velocity v_{max} = 3 cells/time step (corresponding to 81 km/h and thus a reasonable value) for simplicity. The rules for updating the position x of a car are as follows. (i) Acceleration: $v_i = min(v_i + 1, v_{max})$. (ii) Deceleration: $v_i^{'} = min(v_i, g_i)$ so as to avoid collisions, where g_i is the spacing in front of the ith vehicle. (iii) Random brake: with a certain probability p that $v_i^{''} = max(v_i^{'} - 1, 0)$. (iv) Movement: $x_i = x_i + v_i^{''}$.

The fundamental diagram characterizes the basic properties of the NS model which has two regimes called "free-flow" phase and "jammed" phase. The critical density, basically depending on the random brake probability p, divides the fundamental diagram to these two phases. The transition such as from "free-flow" phase to "jammed" phase is called transition on a fundamental diagram (Nagel & Schreckenberg, 1992).

B. Two-route scenario

Recently, Wahle *et al.* (2000) investigated a two-route model. In their model, a percentage of drivers (referred to as dynamic drivers) choose one of the two routes according to the real-time information displayed on the roadside. In their model, the two routes A and B are of the same length L. A new vehicle will be generated at the entrance of the traffic system at each time step. If a driver is a so-called static one, he enters a route at random ignoring any advice. The density of dynamic and static travelers are S_{dyn} and $1 - S_{dyn}$, respectively. Once a vehicle enters one of two routes, the motion of it will follow the dynamics of the NS model. In our simulation, a vehicle will be removed after it reaches the end point. It is important to note that if a vehicle cannot enter the preferred route, it will wait till the next time step rather than entering the un-preferred route.

The simulations are performed by the following steps: first, we set the routes and boards empty; second, let vehicles enter the routes randomly during the initial 100 time steps; third, after the vehicles enter the routes, according to four different feedback strategies, information will be generated, transmitted, and displayed on the board at each time step. Finally, the dynamic road users will choose the route with better conditions according to the dynamic information at the entrance of two routes.

C. Related definitions

The road conditions can be characterized by the fluxes of two routes. The flux of the ith route is defined as follows:

$$F_i = V^i_{mean}\rho_i = V^i_{mean}\frac{N_i}{L_i} \tag{5}$$

where L_i represents the length of the ith route, V^i_{mean} and N_i denote the mean velocity of all the vehicles and the vehicle number on the ith route, respectively. In this chapter, the physical sense of flux F is the number of vehicles passing the exit of the traffic system each time step. Therefore the larger the value of F, the better processing capacity the traffic system has.
We assume the two-route system has only one entrance and one exit as shown in Fig.3. In reality, there are different paths for drivers to choose from one place to another place. In this chapter, we focus on a two-route system. Different drivers departing from the same place could choose two different paths to get to the same destination which corresponds to the "one entrance and one exit" system. Thus the road condition in present work is closer to reality than some previous works (Wahle et al.,2000; Lee et al., 2001; Wang et al., 2005). The rules at the exit of the two-route system are as follows:

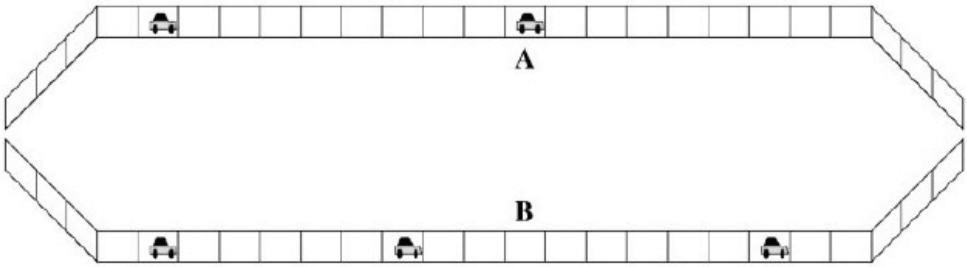


Fig. 3. The one entrance and one exit two-route traffic system.

a) The special velocity update mechanism for the vehicle nearest to the exit:
   i velocity(t)=Min(velocity(t)+1,3), (probability: 75%);
   ii velocity(t)=Max(velocity(t)-1,0), (probability: 25%);
b) Rules at the exit when vehicles competing for driving out:
   i At the end of two routes, the vehicle nearer to the exit goes first.
   ii If the vehicles at the end of two routes have the same distance to the exit, the faster a vehicle drives, the sooner it goes out.
   iii If the vehicles at the end of two routes have the same speed and distance to the exit, the vehicle in the route which has more vehicles drives out first.
   iv If the rules (i), (ii) and (iii) are satisfied at the same time, then the vehicles go out randomly.
c) velocity(t)=position(t)-position(t-1), where position(t)=L=2000; (valid only for the vehicles failed in competing for driving out at exit);

Here we want to stress that the vehicle nearest to the exit will not obey the NS mechanism but the special mechanism as shown in rule (a). However, vehicles following the vehicle closest to the exit still obey the NS mechanism. One should also be aware that if the vehicle nearest

to the exit does not compete with the vehicle on the other route for driving out or wins in the competition, the vehicle will ignore rule (c). The special velocity update mechanism (rule (a)) is equivalent to the situation that 75% drivers exhibit aggressive behavior and 25% drivers exhibit timid behavior near the exit, which is similar to the recent work studied by Laval & Leclercq (2010). Please note that drivers exhibit timid behavior may also exhibit aggressive behavior at next time step otherwise the timid drivers may stop at the exit all the time. Then we describe six different feedback strategies as follows:
TTFS: The information of travel time on the board is set to be zero until one car leave the traffic system. Each vehicle will be recorded the time when it enters and leaves one of the routes. We use the difference between this two values as the feedback information. A new dynamic driver will choose the road with shorter time shown on the information board.
MVFS: Every time step, the traffic control center will receive the velocity of each vehicle on the route from GPS. They will deal with the information and display the mean velocity of vehicles on each route on the information board. Road users at the entrance will choose one road with larger mean velocity.
CCFS: Every time step, the traffic control center will receive the position of each vehicle on the route from GPS. The work of the traffic control center is to compute the congestion coefficient of each route and display it on the information board. Road users at the entrance will choose one road with smaller congestion coefficient. The congestion coefficient is defined as

$$C = \sum_{i=1}^{q} n_i^w. \tag{6}$$

where n_i stands for vehicle number of the ith congestion cluster in which cars are close to each other without a gap between any two of them, and q denotes the total number of congestion clusters on one route. Every cluster is evaluated by a weight w, where $w = 2$ and one can check out that $w > 2$ leads to the similar results with $w = 2$ (Wang et al., 2005).
WCCFS: Every time step, the traffic control center will receive data from the navigation system (GPS) like CCFS, and the work of the center is to compute the congestion coefficient of each road with a reasonable weighted function and display it on the information board. Road users at the entrance will choose one road with smaller weighted congestion coefficient. The weighted congestion coefficient is defined as Eq.(2).
After we try some functions such as $F(x) = \cos(ax) + b$ and Gaussian function, we find $F(x) = kx + b$ is the optimal one in terms of improving the capacity of the road. Here, we set $b \neq 0$ for the reason that it will cause the absolute weight value of the first route site always to be the smallest when $b = 0$. In this chapter, we set $b = 2.0$. Then we get the function as follows:

$$F(x) = k \times x + b = k \times \frac{n_m}{2000} + 2.0. \tag{7}$$

Finally the expression of C_w becomes

$$C_w = \sum_{i=1}^{q} F(n_m) n_i^2 = \sum_{i=1}^{q} (k \times \frac{n_m}{2000} + 2.0) \times n_i^2. \tag{8}$$

We also find that how efficient the new strategy to improve the road capacity depends on the value of the weight factor (slope - k) which we will discuss in detail in Section 3.
CAFS: Every time step, the traffic control center will receive data from the navigation system (GPS) like CCFS. The work of the traffic control center is to compute the corresponding angle

of each congestion cluster (see Fig.2) on the lane, sum square of each corresponding angle up and display it on the information board. Road users at the entrance will choose one road with smaller corresponding angle coefficient. The corresponding angle coefficient is defined as Eq.(3).

PFS: Every time step, the traffic control center will receive data from the navigation system (GPS) like CCFS. The work of the center is to compute the congestion coefficient of each route, simulate the future road condition based on the current road condition by using CCFS, and display the results on the information board. Road users at the entrance will choose one route with smaller predicted congestion coefficient. For example, if the prediction time, T_p, is 50 seconds and the current time is the 100th second, the traffic control center will simulate the road conditions in the next 50 seconds adopting CCFS, predict the road condition at the 150th second, and show the result on the information board at the entrance of the route. Finally, road users at the 100th second will choose one route with smaller predicted congestion coefficient at the 150th second. By the same token, road users at the entrance at the 101th second will choose one route with smaller predicted congestion coefficient at the 151th second like explained above. The predicted congestion coefficient is defined as Eq.(4).

In the following section, performance by using six different feedback strategies will be shown and discussed in detail.

3. Simulation results

Fig.4 (a) shows the dependence of average flux on weight factor (k) by using WCCFS. As to the routes' processing capacity, we can see that in Fig.4 (a) there is a positive peak structure at the vicinity of $k \sim$ -1.98. Thus we will use k=-1.98 in the following paragraphs. Then Eq.(8) will become $C_w = \sum_{i=1}^{q}(-1.98 \times \frac{n_m}{2000} + 2.0) \times n_i^2$. In Fig.4 (b), we present the weight value of each site on one route. One can find the weight value of the entrance is much larger than that of the exit when using WCCFS and the reasons can be described as follows. First, in practice both acceleration and deceleration shock waves travel at final velocities and – at least on average – vehicles tend to be more affected by local traffic conditions than by conditions far ahead on the roadway. This suggests that greater weight should be given to traffic conditions on the upstream sections of each route. Second, the smaller weight value at the end of the route will alleviate the negative effect of congestion caused by the traffic jam.

Since Fig.4 (a) shows the weight value of the entrance is larger than that of the exit when adopting WCCFS, the point T located above the entrance of the route when adopting CAFS (see Fig.2) is reasonable. It makes the weight of the entrance the largest. Furthermore, the corresponding angle of each congestion cluster can reflect not only the weight of the route but also the length of the congestion cluster. Therefore the weight value is more reasonable than before (Dong et al., 2010a). Fig.4 (c) shows the dependence of average flux on position of the pillar (point T) by using CAFS. As to the routes' processing capacity, we can see that the position of the pillar will directly affect the average flux. The average flux is much larger when point T locates at the entrance of the route while the value is pretty lower when point T locates at the end of the lane. Thus the result shown by Fig.4(c) is in accordance with that indicated by Fig.4 (a). Also, this can be understood as shown in Fig.5, where the congestion cluster on route A locates at the entrance of lane and the congestion cluster on route B locates at the end of the lane. From Fig.5, we can see clearly that C_θ of route A is larger than that of route B, so the road user should enter route B instead of route A. If point T locate at the end of the route, C_θ of route A will smaller than that of route B, which will cause the vehicle to enter route A. It will make the cluster larger or the vehicle even cannot enter the route.

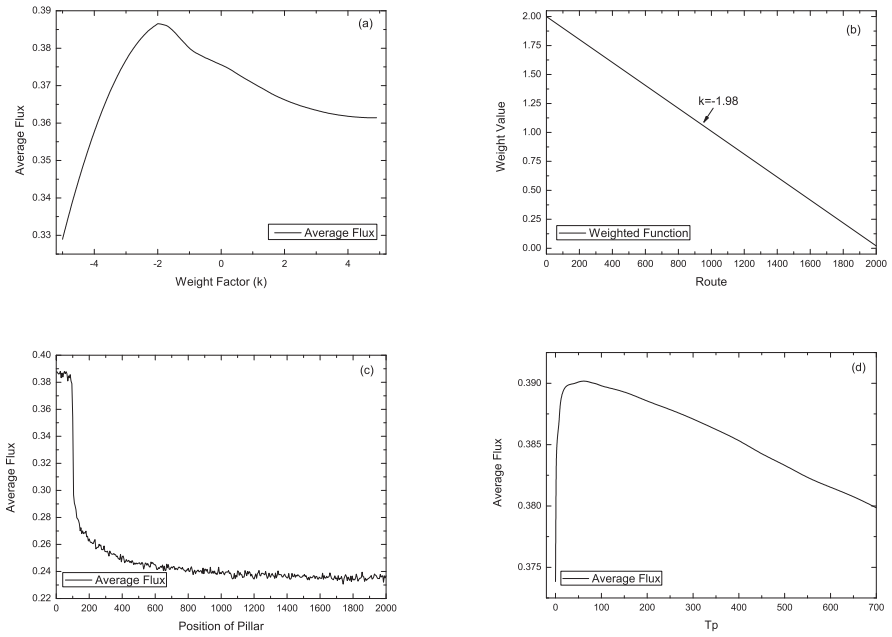


Fig. 4. (a) Average flux *vs* weight factor (k). The parameters are L=2000, p=0.25, and S_{dyn}=0.5. (b) Weight value of each site on one route. The parameters are L=2000, p=0.25, and S_{dyn}=0.5. (c) Average flux *vs* position of pillar (point T). The parameters are L=2000, p=0.25, S_{dyn}=1.0 and vertical distance (H) is fixed to be 100. (d) Average flux *vs* prediction time (T_p). The parameters are L=2000, p=0.25, and S_{dyn}=0.5.

Fig.4 (d) shows the dependence of average flux on prediction time (T_p) by using PFS. As to the routes' processing capacity, we can see that in Fig.4 (d) there is a positive peak structure at the vicinity of $T_p \sim 60$. Thus we will use T_p=60 in the following paragraphs. PFS is more difficult to realize than other five feedback strategies since PFS is based on the future road condition. As demonstrated by our previous work (Dong et al., 2010), more routes the traffic system has, longer prediction time (T_p) PFS needs (see Fig.6). The workload adopting CCFS is equivalent to the workload adopting PFS when $T_p = 0$ (which is equivalent to CCFS). Thus in a two-route scenario, the workload of PFS is sixty times greater than that of CCFS because the prediction time equal to 60 time steps ($T_p = 60$). Given the time interval of every time feedback is one time step, there is no doubt that PFS requires higher performance computers to operate than the other five feedback strategies. It indicates that PFS will cost more to realize than others.

Fig.7 shows simulation results of applying TTFS in a two-route scenario with respect to flux, number of cars, and average speed all versus time step. The fluxes of two routes adopting TTFS show oscillation (see Fig.7) obviously due to the information lag effect (Lee et al., 2001). This lag effect can be understood. For TTFS, the travel time reported by a driver at the end of two routes only represents the road condition in front of him, and perhaps the vehicles behind him have got into the jammed state. Unfortunately, this information will induce more vehicles

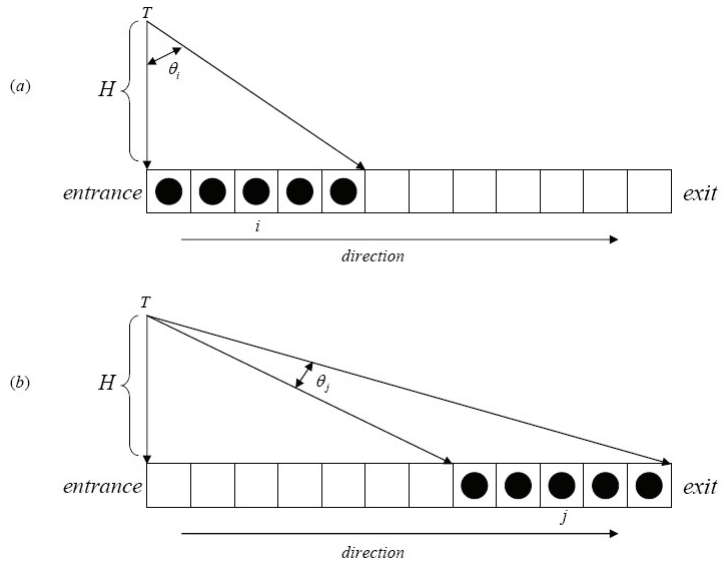


Fig. 5. The locations and corresponding angles of vehicle congestion clusters on route A and route B.

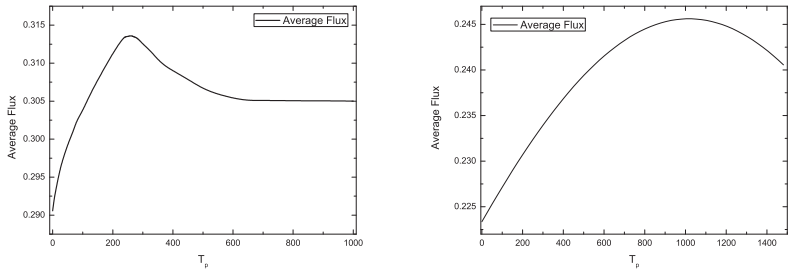


Fig. 6. (a) Average flux *vs* prediction time (T_p) in a three-route traffic system and $T_p(F_{max}) \sim$ 260. (b) Average flux *vs* prediction time (T_p) in a four-route traffic system and $T_p(F_{max}) \sim$ 1020. The parameters are L=2000, p=0.25, and S_{dyn}=0.5.

to choose his route until a vehicle from the jammed cluster leaves the system. This effect apparently does harm to the system. Another reason for the oscillation is that the two-route system only has one exit and the vehicle nearest to the exit obeys the special velocity update mechanism; therefore at most one vehicle can go out each time step. It will result in the traffic jam happening at the end of the route. Vehicle number versus time step shows almost the same tendency as flux versus time step (see Fig.7 (b)) and the average velocity is around 2.4 cells per time step (refer to Fig.7(c)).

Fig.8 shows simulation results of applying MVFS in a two-route scenario with respect to flux, vehicle number, and average speed all versus time step. The fluxes of two routes adopting

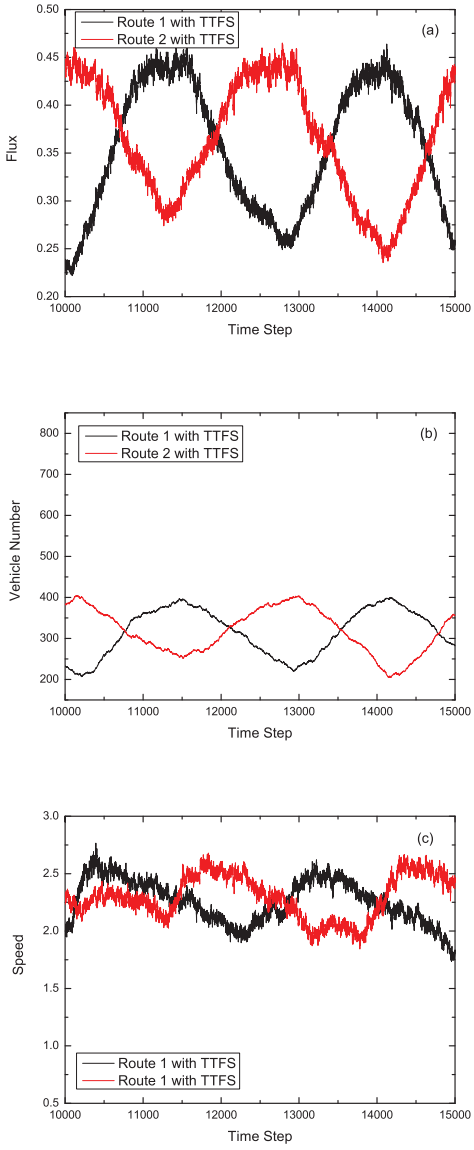


Fig. 7. (Color online)(a) Flux of each route with travel time feedback strategy. (b) Vehicle number of each route with travel time feedback strategy. (c) Average speed of each route with travel time feedback strategy. The parameters are L=2000, p=0.25 and S_{dyn}=0.5.

MVFS showing oscillation (see Fig.8) is primarily due to two reasons. First, for MVFS, we have mentioned that the NS model has a random brake scenario which causes the fragile stability of velocity, thus MVFS cannot completely reflect the real condition of routes. The other reason for the disadvantage of MVFS is that flux consists of two parts, mean velocity and vehicle density, but MVFS only grasps one part and lacks the other part of flux (Wang et al., 2005). Also, the one exit structure of the traffic system and the special velocity update mechanism for the vehicle closest to the exit can also cause oscillation as explained in the last paragraph. Vehicle number versus time step shows almost the same tendency as flux versus time step (see Fig.8 (b)) and the average velocity is around 2.3 cells per time step (refer to Fig.8(c)).
Fig.9 show the dependence of flux, number of cars, average speed on time step by using CCFS. Compared with TTFS and MVFS, the performance of CCFS is good. The reason is primarily due to that CCFS takes the congestion cluster effects into account by adding a weight to each cluster. This can be explained by the fact that travel time of the last vehicle of the cluster from the entrance to the destination is obviously affected by the size of cluster. With the increasing of cluster size, travel time of the last vehicle will be longer, and the correlation between cluster size and travel time of the last vehicle is nonlinear. For simplicity, an exponent w is added to the size of each cluster to be consistent with the nonlinear relationship. To some extent, CCFS reduces the oscillation, and increases the vehicle number of each route (see Fig.9 (a) & (b)) while decreases the average velocity that is approximate to 2.2 cells per time step (refer to Fig.9(c)).
The dependence of flux, number of vehicles, average speed on time step by adopting WCCFS is shown in Fig.10. WCCFS further reduces the oscillation and increases the flux due to the fact that WCCFS takes the weights of different parts of the route into consideration. From Fig.4 (b) we can see that weight values at the end of the route are always smaller, which is equivalent to alleviating the negative effect of congestion caused by the traffic jam; therefore WCCFS may improve the road condition. Compared to CCFS, the performance adopting WCCFS is improved at some points, not only on the value but also the stability of the flux.
Fig.11 shows the relationship between flux, number of vehicles, average speed and time step by using CAFS. In contrast with CAFS, the fluxes of two routes adopting TTFS, MVFS, CCFS and WCCFS show larger oscillation (see Fig.7-10). This oscillation effect can be understood for several reasons besides those discussed above. First, TTFS, CCFS and MVFS cannot reflect the weights of different parts of the route. Additionally, though WCCFS can reflect the route weights, the weighted function ($F(n_m)$) is independent of the cluster length. We use the median rounding $\lfloor n_m \rfloor$ of the ith congestion cluster to represent its position when adopting WCCFS. However, CAFS takes both the length and location of the congestion cluster into account, which can give the road user with better guidance. For example, if there exit congestion clusters at the end of both routes, the road user will choose the route with shorter cluster length. The reason is that there is a positive correlation between the value of C_θ and the length of the cluster when the locations of clusters are the same. If the clusters have the same length but locate at different positions of the routes as shown in Fig.5, the road user will choose route B with smaller C_θ. Compared to WCCFS, the performance adopting CAFS is further improved, not only on the value but also the stability of the flux.
In Fig.11 (b), vehicle number versus time step shows that the routes' accommodating capacity is greatly enhanced with an increase in average vehicle number. Thus perhaps the high fluxes of two routes with CAFS are mainly due to the increase of vehicle number. In Fig.11 (c), speed versus time step shows that although the speed is more stable by using CAFS, it becomes lower than the other four strategies discussed above. The reason is that the routes'

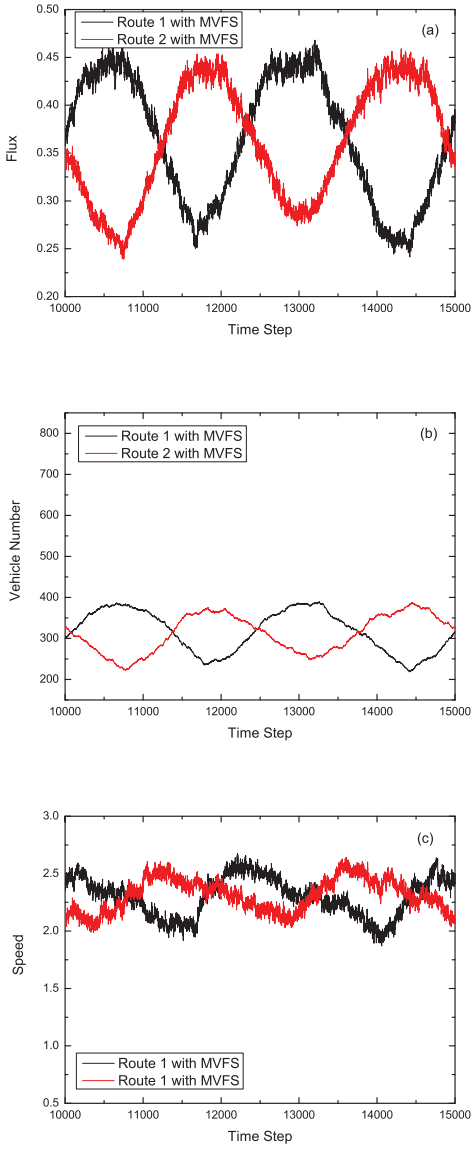


Fig. 8. (Color online)(a) Flux of each route with mean velocity feedback strategy. (b) Vehicle number of each route with mean velocity feedback strategy. (c) Average speed of each route with mean velocity feedback strategy. The parameters are L=2000, p=0.25 and S_{dyn}=0.5.

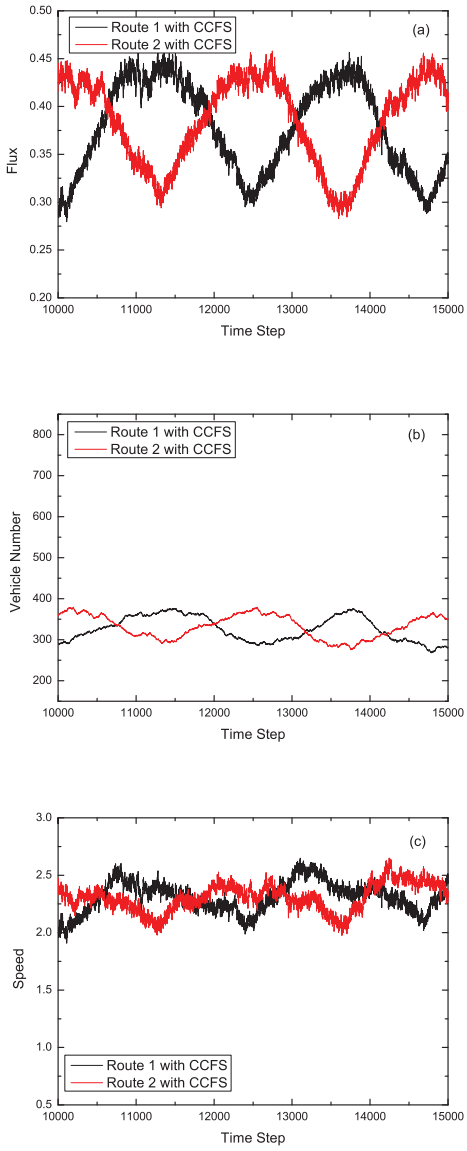


Fig. 9. (Color online)(a) Flux of each route with congestion coefficient feedback strategy. (b) Vehicle number of each route with congestion coefficient feedback strategy. (c) Average speed of each route with congestion coefficient feedback strategy. The parameters are L=2000, p=0.25 and S_{dyn}=0.5.

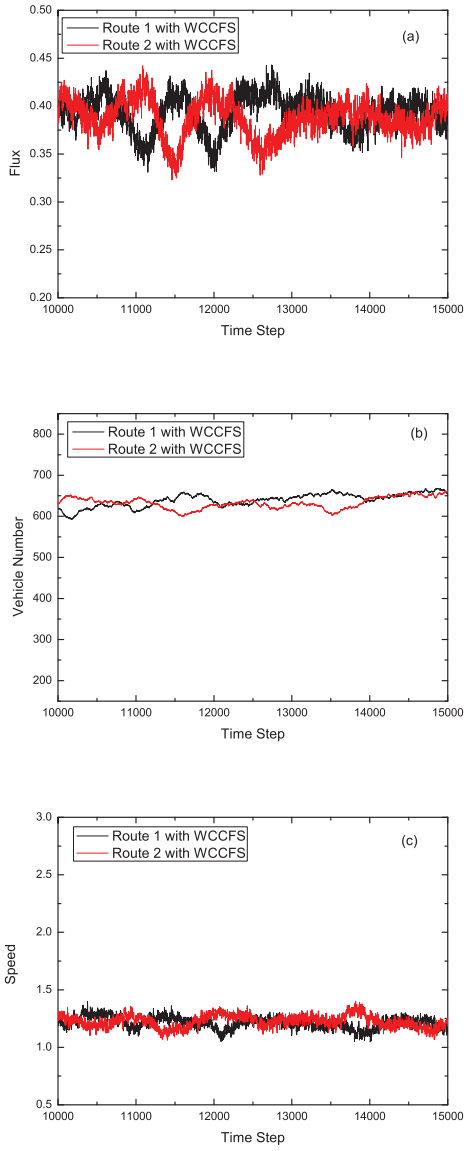


Fig. 10. (Color online)(a) Flux of each route with weighted congestion coefficient feedback strategy. (b) Vehicle number of each route with weighted congestion coefficient feedback strategy. (c) Average speed of each route with weighted congestion coefficient feedback strategy. The parameters are L=2000, p=0.25, S_{dyn}=0.5, and weight factor (k) is fixed at -1.98.

accommodating capacity by using CAFS is better than that of other four strategies, and at most one vehicle can drive out each time step as explained before; therefore more cars the lane has, lower speeds the vehicles have. Fortunately, flux consists of two parts, mean velocity and vehicle density. Hence, as long as the vehicle number (because vehicle density is defined as ρ=N/L, and L is fixed at 2000, so $\rho \propto$ vehicle number (N)) is large enough, the flux can also be the largest.

Fig.12 shows the dependence of flux, vehicle number, average speed on time step when adopting PFS. The advantage of PFS is that it can predict the negative effects on the route condition caused by traffic jams happened at the end of the route, try to avoid jammed state to the best of its ability, and alleviate the negative effects as much as possible. Here, we want to stress that though PFS try to avoid the jammed state, the structure of the traffic system (one exit) and the special velocity update mechanism for the vehicle nearest to the exit still make jams happened at the end of the route occasionally. Also, this can explain the slight oscillation in Fig.12 (a).

From Fig.12 (b) we know that the average vehicle number is around 760 by using PFS. As to the routes' stability, we know PFS is the optimal one, which means the vehicles should be almost uniformly distributed on each route instead of being together at the end of the routes. Furthermore, even there are 760 vehicles on each route, vehicles can still averagely occupy $2 \sim 3$ sites on each route because the total length of each route is fixed at 2000 sites. This indicates there are only $1 \sim 2$ sites between vehicles. Though the vehicles are almost distributed separately on each lane, the one exit structure and the special velocity update mechanism for the vehicle closest to the exit make jams still have a chance to happen at the end of the route. However, PFS can prevent jams from further expanding and alleviate the negative effects as much as possible, so that the jammed state will disappear soon. So as to analogize, even the jams happen again, the poor road condition will be relieved in a short time period.

As to the low speed shown in Fig.12 (c), the reason is that the speed partially depends on the number of empty sites between two vehicles on the lane. The vehicle behind another vehicle can move at most the current empty sites between them which is required by NS mechanism (Nagel & Schreckenberg, 1992). The routes' accommodating capacity is the best by using PFS, indicating the speed adopting PFS the lowest. From the stability of the velocity, we infer that the vehicles should drive at almost uniform speeds on each route. Without consider other factors, the speed should be a little more than one (~ 1.5) because there are only $1 \sim 2$ cells between vehicles as mentioned above. If we take the random brake effects and the occasional jams at the end of the route into account, the vehicles' average velocity decreasing a little is possible and reasonable. Thus, the average velocity $V_{avg} \sim 1$ in this chapter could be understood. These analysis can also be applied to explain the low average velocity by using WCCFS and CAFS.

Someone may have doubts whether CAFS and PFS are really better than the other four feedback strategies due to the lower speed shown in Fig.11 (c) & Fig.12 (c). In order to making the readers understand more easily, we assume that the road network under study includes not only the downstream corridor section displayed in Fig.3, but also a section upstream of the entrance where vehicles wait to enter the corridor (Dong et al., 2010c). Thus, we should take into account the total travel time (t_{tot}) that is the sum of driving time ($t_{driving}$) and waiting time ($t_{waiting}$) to evaluate the merits of these feedback strategies.

$$t_{tot} = t_{driving} + t_{waiting} \tag{9}$$

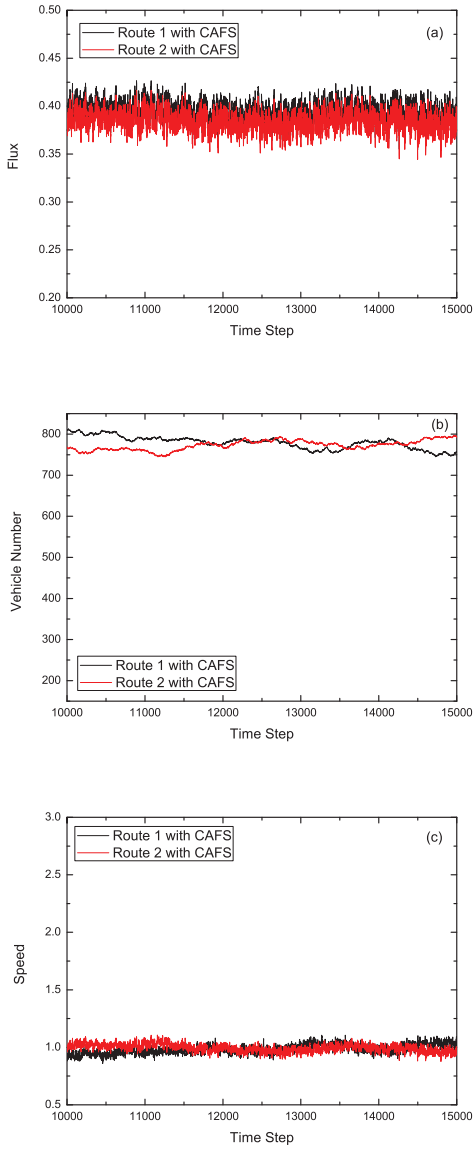


Fig. 11. (Color online)(a) Flux of each route with corresponding angle feedback strategy. (b) Vehicle number of each route with corresponding angle feedback strategy. (c) Average speed of each route with corresponding angle feedback strategy. The parameters are L=2000, p=0.25, S_{dyn}=0.5, and vertical distance (H) is fixed at 100.

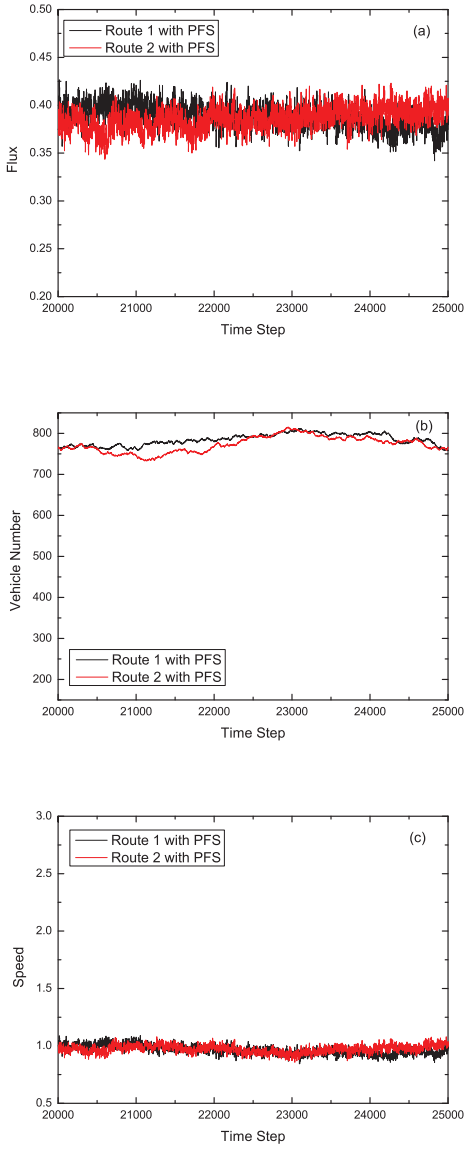


Fig. 12. (Color online)(a) Flux of each route with prediction feedback strategy. (b) Vehicle number of each route with prediction feedback strategy. (c) Average speed of each route with prediction feedback strategy. The parameters are L=2000, p=0.25, S_{dyn}=0.5, and T_p=60.

Total travel time (t_{tot}): t_{tot} is the time period that vehicles spend on the whole traffic system that includes the upstream and downstream corridor sections. Flux is a very good proxy to evaluate the total travel time, because it can be understood as the number of vehicles passing the exit each time step. The more vehicles pass the exit during a fixed time period, the shorter time these vehicles spend on the traffic system. For example, there are totally N vehicles in the section upstream of the entrance, then the average flux of the traffic system is

$$F_{avg} = N/t_{tot\ (Nth\ vehicle)} \approx n/t_{tot\ (nth\ vehicle)},\ n \in [1, N]. \tag{10}$$

Here, one should be aware that the average flux value is stable, thus the approximation in Eq.(10) is valid. Therefore the travel time adopting CAFS and PFS is shorter than the other four feedback strategies for the fact that the flux adopting CAFS and PFS are larger (see Fig.13).
Driving time ($t_{driving}$): $t_{driving}$ is the time period vehicles spend on the downstream route section displayed in Fig.3. It is obvious that driving time of CAFS and PFS is longer than the other four feedback strategies since the speeds v adopting CAFS and PFS are lower as shown in Fig.11 (c) & Fig.12 (c) (the length of the route L is fixed at 2000, thus $t_{driving} = L/v$).
Waiting time ($t_{waiting}$): $t_{waiting}$ is the time period vehicles waiting in the upstream route section. Since the total travel time (t_{tot}) is the sum of waiting time ($t_{waiting}$) and driving time ($t_{driving}$) as shown in Eq.(9), the waiting time adopting CAFS and PFS is shorter on the basis of above analysis.
Fig.13 shows that the average flux fluctuates feebly with a persisting increase of dynamic travelers by using six different strategies. As to the routes' processing capacity, PFS is proved to be the best one because the flux is always the largest at each S_{dyn} value and even increases with a persisting increase of dynamic travelers.

4. Conclusion

We obtain the simulation results of applying six different feedback strategies, i.e., TTFS, MVFS, CCFS, WCCFS, CAFS and PFS in a two-route scenario all with respect to flux, number of vehicles, speed, and average flux versus S_{dyn}. We also show the results about average flux versus weight factor (k) by adopting WCCFS, average flux versus position of pillar (point T) by adopting CAFS, and average flux versus prediction time (T_p) by adopting PFS. These results indicate that PFS has more advantages than the other five strategies in the two-route system with only one entrance and one exit. However, as we stress before that PFS is not easy to realize and will be invalid when the transportation system is multi-route (Dong et al., 2010d). The numerical simulations demonstrate that the weight factor k (WCCFS), the position of point T (CAFS) and the prediction time T_p (PFS) play very important roles in improving the road conditions. In contrast with other four feedback strategies (TTFS, MVFS, CCFS and WCCFS), CAFS and PFS can significantly improve the road conditions, including increasing vehicle number and flux, reducing oscillation, and enhancing average flux with the increase of S_{dyn}. This can be understood because CAFS takes both the length and location of each congestion cluster into consideration; and PFS can predict the future road conditions.
With the development of science and technology, it is not difficult to realize these advanced information feedback strategies in reality. The position and velocity information of vehicles will be known through the navigation system (GPS). Then these feedback strategies can come true through computational simulation. Though the performance of PFS is the optimal one, it will cost most when adopting PFS as explained before. The rest five feedback strategies should cost almost the same. If someone can propose one feedback strategy in the near future whose

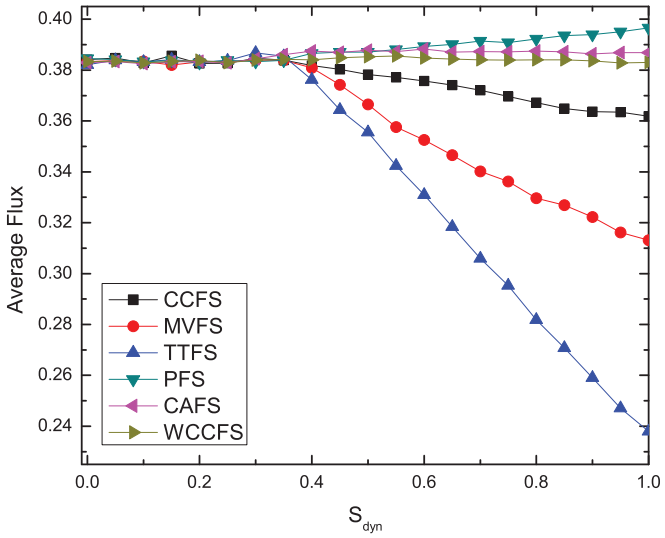


Fig. 13. (Color online) Average flux by performing different strategies *vs* S_{dyn}; L is fixed at 2000, p is fixed at 0.25.

performance is better than PFS and cost is similar to CAFS, it will make great contributions to radically improve the road conditions of real-time traffic systems.

5. Acknowledgments

This work has been partially supported by the Georgia Institute of Technology, the National Basic Research Program of China (973 Program No. 2006CB705500), the National Natural Science Foundation of China (Grant Nos. 10975126 and 10635040), the National Important Research Project: (Study on emergency management for non-conventional happened thunderbolts, Grant No. 91024026), and the Specialized Research Fund for the Doctoral Program of Higher Education of China (Grant No.20093402110032).

6. References

Adler, J. L. and Blue, V. J. (1998). Toward the design of intelligent traveler information systems. Transportation Research Part C, 6:157-172.

Arnott, R., de Palma, A., and Lindsey, R. (1991). Does providing information to drivers reduce traffic congestion. Transportation Research Part A, 25:309-318.

Barlovic, R., Santen, L., Schadschneider, A., Schreckenberg, M. (1998). Metastable states in cellular automata for traffic flow. European Physical Journal B , 5:793-800.

Ben-Akiva, M., de Palma, A., and Kaysi, I. (1991). Dynamic network models and driver information-systems. Transportation Research Part A, 25:251-266.

Biham, O., Middleton, A. A., and Levine, D. (1992). Self-organization and a dynamic transition in traffic-flow models. Physical Review A, 46:R6124-R6127.

Blue, V. J. and Adler, J. L. (2001). Cellular automata microsimulation for modeling bi-directional pedestrian walkways. Transportation Research Part B, 35:293-312.

Chowdhury, D., Santen, L., and Schadschneider, A. (2000). Statistical physics of vehicular traffic and some related systems. Physics Reports, 329:199-329.

Dong, C. F. et al. (2009a). "Intelligent Traffic System Predicts Future Traffic Flow on Multiple Roads." PHYSorg.com. 12 Oct 2009. http://www.physorg.com/news174560362.html.

Dong, C. F. and Ma, X. (2010a). Corresponding Angle Feedback in an innovative weighted transportation system. Physics Letters A, 374:2417–2423.

Dong, C. F., Ma, X., and Wang, B. H. (2010b). Weighted congestion coefficient feedback in intelligent transportation systems. Physics Letters A, 374:1326–1331.

Dong, C. F., Ma, X., and Wang, B. H. (2010c). Effects of vehicle number feedback in multi-route intelligent traffic systems. International Journal of Modern Physics C, 21:1081-1093.

Dong, C. F., Ma, X., Wang, B. H., and Sun, X. Y. (2010d). Effects of prediction feedback in multi-route intelligent traffic systems. Physica A, 389:3274-3281.

Dong, C. F., Ma, X., Wang, G. W., Sun, X. Y., and Wang, B. H. (2009b). Prediction feedback in intelligent traffic systems. Physica A, 388:4651-4657.

Friesz, T. L., Luque, J., Tobin, R.L., and Wie, B. W. (1989). Dynamic network traffic assignment considered as a continuous-time optimal-control problem. Operations Research, 37:893-901.

Fu, C. J., Wang, B. H., Yin, C. Y., Zhou, T., Hu, B., and Gao K. (2007). Analytical studies on a modified Nagel-Schreckenberg model with the Fukui-Ishibashi acceleration rule. Chaos, Solitons and Fractals, 31:772-776.

Fukui, M., Ishibashi, Y. (1996). Traffic flow in 1D cellular automaton model including cars moving with high speed. Journal Of The Physical Society Of Japan, 65:1868-1870.

Gao, K., Jiang, R., Hu, S. X., Wang, B. H., and Wu, Q. S. (2007). Cellular-automaton model with velocity adaptation in the framework of Kerner's three-phase traffic theory. Physical Review E, 76:026105.

Helbing, D. (1996). Traffic and Granular Flow, chapter Wolf, D.E., Schreckenberg, M., and Bachem, A., editors, Traffic modeling by means of physical concepts, pages 87-C104. World Scientific Publishing.

Helbing, D. and Treiber, M. (1998). Gas-kinetic-based traffic model explaining observed hysteretic phase transition. Physical Review Letters, 81:3042-3045.

Helbing, D. (2001). Traffic and related self-driven many-particle systems. Reviews of Modern Physics, 73:1067-1141.

Hu, S. X., Gao, K., Wang, B. H., Lu, Y. F., and Fu C. J., (2007). Abnormal hysteresis effect and phase transitions in a velocity-difference dependent randomization CA model. Physica A, 386:397-406.

Jiang, R., Wu, Q. S. (2003). Cellular automata models for synchronized traffic flow. Journal of Physics A, 36:381-390.

Jiang, R., Wu, Q. S. (2005). First order phase transition from free flow to synchronized flow in a cellular automata model. European Physical Journal B, 46:581-584.

Kachroo, P. and Özbay, K. (1996). Real time dynamic traffic routing based on fuzzy feedback control methodology. Transportation Research Record, 1556:137-146.

Knospe, W., Santen, L., Schadschneider, A., Schreckenberg, M. (2000). Towards a realistic microscopic description of highway traffic. Journal of Physics A, 33:L477-L485.

Kerner, B. S. (2004). Three-phase traffic theory and highway capacity. Physica A, 333:379-440.

Kerner, B. S., Klenov, S. L. (2002). A microscopic model for phase transitions in traffic flow. Journal of Physics A, 35:L31-L43.

Kerner, B. S., Klenov, S. L., and Wolf, D. E. (2002). Cellular automata approach to three-phase traffic theory. Journal of Physics A, 35:9971-10013.

Laval, J. A., and Leclercq. L. (2010). A mechanism to describe the formation and propagation of stop-and-go waves in congested freeway traffic. Philosophical Transactions Of The Royal Society A-Mathematical Physical And Engineering Sciences, 368:4519-4541.

Lee, K., Hui, P. M., Wang, B. H., and Johnson, N. F. (2001). Effects of announcing global information in a two-route traffic flow model. Journal of the Physical Society of Japan, 70:3507-3510.

Li, X. B., Wu, Q. S., Jiang, R. (2001). Cellular automaton model considering the velocity effect of a car on the successive car. Physical Review E, 64:066128.

Mahmassani, H. S. and Jayakrishnan, R. (1991). System performance and user response under real-time information in a congested traffic corridor. Transportation Research Part A, 25:293-307.

Mao, D., Wang, B. H., Wang, L., Hui, R. M. (2003). Traffic flow CA model in which only the cars following the trail of the ahead car can be delayed. International Journal of Nonlinear Science and Numerical Simulation, 4:239-250.

Nagatani, T. (2002). The physics of traffic jams. Reports on Progress in Physics, 65:1331-1386.

Nagel, K. and Schreckenberg, M. (1992). A cellular automaton model for freeway traffic. J. Phys. I, 2:2221-2229.

Paveri-Fontana, S.L. (1975). Boltzmann-like treatments for traffic flow - critical review of basic model and an alternative proposal for dilute traffic analysis. Transportation Research, 9:225-235.

Prigogine, I. and Andrews, F. C. (1960). A Boltzmann-like approach for traffic flow. Operations Research, 8:789-797.

Rothery, R. W. (1992). Gartner, N., Messner, C. J., and Rathi, A.J., editors, Traffic Flow Theory. Transportation Research Board: Transportation Research Board Special Report, 165: Chapter 4. Washington, D.C.

Wahle, J., Bazzan, A. L. C., Klügl, F., and Schreckenberg, M. (2000). Decision dynamics in a traffic scenario. Physica A 287:669-681.

Wahle, J., Bazzan, A. L. C., Klügl, F., and Schreckenberg, M. (2002). The impact of real-time information in a two-route scenario using agent-based simulation, Transportation Research Part C, 10:399-417.

Wang, B. H., Mao, D., and Hui, P. M. (2002). The two-way decision traffic flow model: Mean field theory. In: *Proceedings of The Second International Symposium on Complexity Science, Shanghai, August 6-7*, pages 204-211.

Wang, B. H., Mao, D., Wang, L., Hui, P. M. (2003). Traffic and Granular Flow'01, Fukui, M., Sugiyama, Y., Schreckenberg, M., and Wolf, D. E., editors, Spacing-Oriented Analytical Approach to a Middle Traffic Flow CA Model Between FI-Type and NS-Type, pages 51-64. Springer-Verlag Press.

Wang, B. H., Wang L., Hui, P. M., and Hu B. (2000a). The asymptotic steady states of deterministic one-dimensional traffic flow models. Physica B, 279:237-239. 32

Wang, B. H.,Wang L., Hui, P. M., and Hu B. (2000b). CA model for 1-D traffic flow with gradual acceleration and stochastic delay: Analytical approach. International Journal of Nonlinear Science and Numerical Simulation, 1:255-266.

Wang L., Wang, B. H., and Hu B. (2001). Cellular automaton traffic flow model between the Fukui-Ishibashi and Nagel-Schreckenberg models. Physical Review E, 63:056117.

Wang, W. X., Wang, B. H., Zheng, W. C., Yin, C. Y., and Zhou, T. (2005). Advanced information feedback in intelligent traffic systems. Physical Review E, 72:066702.

Yokoya, Y. (2004). Dynamics of traffic flow with real-time traffic information. Physical Review E, 69:016121.

12

Network Systems Modelled by Complex Cellular Automata Paradigm

Pawel Topa
AGH University of Science and Technology
Poland

1. Introduction

Most historical sources say that Cellular Automata were discovered by John von Neumann and Stanislaw Ulam in the forties of the twentieth century. Von Neumann was investigating the problem of self-reproducing systems in biology. His initial kinematic model appeared to be too complicated and he followed the suggestion from Stanislaw Ulam, who at the same time worked on problems of crystal growth. Von Neumann and Ulam defined an abstract universe in the form of two-dimensional regular mesh with interacting entities Neumann (1966).
In the seventies mathematician John Conway published his set of rules called "Game of Life". His work was popularized by Martin Gardner in the pages of *Scientific American*. The rules of "Game of Life" implemented on computers showed an amazing world of living and dying automata. Its dynamics appeared to be so surprisingly complex that it became area of intensive studies. Vast amount of work concentrated on searching for initial conditions that evolve into non-trivial behaviour of Cellular Automata. One of the most important achievement was proving that "Game of Life" can simulate Universal Turing Machine i.e. can act as an abstract model of any computer.
In the eighties Stephen Wolfram spent part of his scientific career on investigating properties of one-dimensional Cellular Automata. His important achievement was definition of 256 rules for one-dimensional Cellular Automata. He also introduced first classification scheme based on long period observation. The crowning achievement of his works was a book "A New kind of Science" Wolfram (2002). In this monumental work Wolfram outlines a new paradigm for modelling complex phenomena with Cellular Automata approach.
Scientific research related to Cellular Automata were not only limited to abstract mathematical "toy". Toffoli, Margoulus and Fredkin postulated application of cellular automata for modelling physical systems Chopard & Droz (1998); Toffoli & Margolus (1987). They noticed that physical laws can be encoded in rules of interaction in temporally and spatially discrete universe.
The milestone in Cellular Automata theory was the lattice gas model HPP developed by Hardy et al. (1976). The model, which is in fact the Cellular Automata, consists of a simple and fully discrete system of moving and colliding particles. Although, the idea of lattice gas automata was invented independently, the Cellular Automata theory provided clear conceptual framework. HPP model and his successor FHP Frisch et al. (1986) was successful in modelling complex phenomena as flows in porous media, spreading of a liquid droplet and wetting phenomena.
It is difficult now to figure out all the areas where the Cellular Automata paradigm is applied:

- biological processes: tissue growth Wang et al. (2008), angiogenesis Markus et al. (1999), cancer development Reis et al. (2009) etc.
- chemical processes Chopard & Droz (1998); Kier (2000); Kier et al. (1998),
- granular flow Baxter & Behringer (1991); Masselot & Chopard (1996),
- pedestrian dynamics and traffic flow Blue et al. (1997); Was (2005),
- geological processes D'Ambrosio et al. (2001).

"Cellular Automata Modelling of Physical Systems" by Chopard & Droz (1998) contains very profound review of various applications of Cellular Automata approach for modelling physical systems.

Cellular Automata in their classic meanings have rather limited application area. On the other hand their simplicity in computer implementation is irresistible temptation. A lot of extensions were introduced to their definition. The set of states was extended by introducing floating point numbers instead of logical values. The neighbourhood notion was extended by introducing new schema (von Neumann, Moore and Margoulus neighbourhoods) and range bigger then one cell. The homogeneity of Cellular Automata were also broken. In some applications synchronous updating were also replaced by other asynchronous schemas. Mesh types other then rectangular were also applied in some models. Some extensions make that model is so far from classic Cellular Automata definition that it could be classified as a different modelling tool. The Cellular Potts model Graner & Glazier (1992) and Agent-Base models Wooldridge (2009) can be treat as generalized Cellular Automata.

Cellular automata paradigm fits perfectly to paradigm of computation on digital computers. When we use Cellular Automata approach we do not have to struggle with round-off errors, truncation errors, numerical stability and lots of other problems that appear when we implement methods of numerical analysis on computers. Rules of local interaction can be easily and unambiguously defined by using programming languages constructions (*if ... then ... , while ... do ...*). Algorithms and data structures necessary to develop Cellular Automata models are relatively simple and give great opportunity to efficient implementation. Moreover, Cellular Automata are inherently parallel. Cellular Automata definition states that cells have to be processed simultaneously. Computer implementation have to simulate this by using additional temporary data structure. In the 80s and 90s of 20th century a lot of work had been devoted to constructing hardware implementation of Cellular Automata. The most famous were Cellular Automata Machines built by Tommaso Toffoli and Norman Margoulus (CAM-6 i CAM-8) Margolus et al. (1986); Toffoli & Margolus (1987). There were also successful attempts to implement CA on transputer machines Cannataro et al. (1995); Somers & Rem (1989) however this architecture did not succeed an fallen into oblivion. Finally, these interesting researches were abandoned in favour of flexibility of programming solutions. Recent research in GPGPU computing methods shows that Cellular Automata can run very effective on graphic processors Gobron et al. (2010).

In this chapter I present applications of Cellular Automata paradigm for modelling dynamically evolving system with networks structure that function as transportation pathways. The following section presents phenomena that was studied and modelled by the author. Next I present the general idea of Graph of Cellular Automata that combine of Cellular Automata and graphs. In further section I present two models that employed Graph Cellular Automata:

- model of anastomosing river system,
- model of tumour induced angiogenesis.

I conclude the chapter with discussion on extensions and modifications that make Cellular Automata useful in modelling physical systems.

2. Network systems modelled with cellular automata

Cellular Automata fit best for modelling systems that are characterized by inner structural homogeneity and regularity, for example crowd of pedestrians or fluids. When we look for Cellular Automata that are able to capture complex systems with interacting parts that have a different structure, we have to extended classic definition and combine it with other methods. One of the phenomena that we observe in everyday life are transportation systems located in consuming or producing environments. Transportation networks are transfer pathways for various substances which are produced or absorbed by surrounding environments. These substances can also diffuse through the environment. We can enumerate two examples of such the systems that we meet in everyday life:

- Vascular systems — blood vessels transport nutrients and oxygen which penetrate into the tissue and nourish it. Products of metabolism penetrate from tissue to blood vessels and they are removed from the body. Tissue can influence the vascular network by producing various substances called angiogenic factors that stimulate angiogenesis (process of formation of blood vessels).
- River systems — rivers transport organic and mineral materials from upper, erosional part of river system to lower part were the loads are deposited. In whole the basin, the surrounding environment (shape of terrain) are modified by river (erosion and deposition). Changes in terrains influence river systems by changing the routes of channels as well as the shape of network.

Such the systems are characterized by coexistence two parts having completely different structure. Environment has (in general) regular, uniform structure where substances are distributed or collected in whole area or volume. Regular mesh with dense network of connection between cells can accurately approximate these processes. In the networks, processes are rather limited to the paths defined by their branches. Thus, we can easily simplify their representation by using graph.
Graph of Cellular Automata is a hybrid modelling method employing Cellular Automata theory and some elements of graph theory. Cellular Automata is a basis that is used to construct graph. Some cells from mesh of automata are selected and they establish set of nodes in graph. An additional relations of neighbourhood are established between these cells, what constitute set of edges. Figure 1 illustrates this structure.

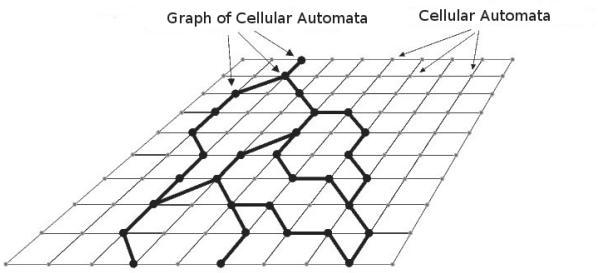


Fig. 1. Graph of Cellular Automata, from Topa & Dzwinel (2009)

Regular mesh and graph are processed separately and often in different manner (e.g. synchronously versus asynchronously). Cells that belong to the graph act as connections between these two systems and provide interaction between them. State of these cells are transfer way for results of environment simulation on regular mesh of Cellular Automata to the graph structure representing network system (see Fig. 2).

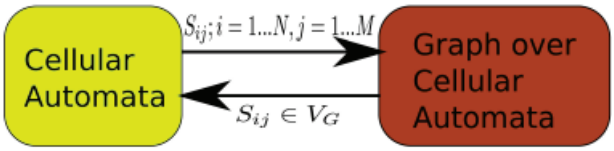


Fig. 2. Interaction between Cellular Automata and Graph constructed over Cellular Automata is transfered through the states of cells. States of all the cells in the mesh (S_{ij}) can influence evolution of network system encoded in graph. Network system influences environment through the cells that belong to graph ($S_{ij} \in V_G$)

The paradigm of Graph of Cellular Automata was applied for simulating two phenomena:

- Anastomosing river systems Topa & Dzwinel (2003); Topa et al. (2006); Topa & Paszkowski (2001)
- Tumour Induced Angiogenesis Topa (2006; 2008); Topa & Dzwinel (2009)

2.1 Rivers and anastomosing river systems

Anastomosing rivers are interesting dynamic system created as a result of interaction between constructive and destructive processes. Anastomosing systems are formed in middle part of river system. Slope on these area is very small — approximately 10cm per 1km. Flow rate is also very small, river do not erode terrain. Organic and non-organic materials carried by river are mostly transported through this part of river and deposited later in lower reaches. Waters rich of nutrients penetrate soil in surrounding banks. Fertile areas near river channels are characterized by a significant increase in vegetation. Products of plant decay deposit on this area as peat (approximately 1-1.5 mm/year Gradzinski et al. (2003)). Peat accumulation lasting for thousands years may produce a peat bog with a depth of several meters.

The processes that occur on river channel banks are crucial for creation anastomosing system. Figure 3 illustrates these processes. It presents cross-section of river channel. The rate of peat accumulation depends on the distance from river channel. Water penetrates soil nearby the channel and supplies nutrients. Nutrients are gathered by plants that vegetate on the river banks. On the areas that are more distant from water, the vegetation are less abundant than on the areas that directly neighbour to channel. Consequently, the rate of peat deposition decreases with distance from channel. Deposition on river banks is accompanied by deposition on river bottoms. As a result the whole river channel is raised (see in Fig 3). The water level is raised above the average valley ground level. Sometimes channel can be also partly blocked by clusters of floating plants, tree trunks etc. It is metastable state which can be easily destroyed. High levels of water that usually occur during spring floods can cause breaking river banks above obstacles. River can create bypass channel that evade narrowness on primary channel. The shape of terrain in valley usually force new channel to join primary channel somewhere downstream.

The routes of new channels run through local depression areas. Evolution of new channel proceed in the same way as the parent channel. Old channels may also disappear gradually. The system evolves to dynamic metastable state in which whole the area of the valley is

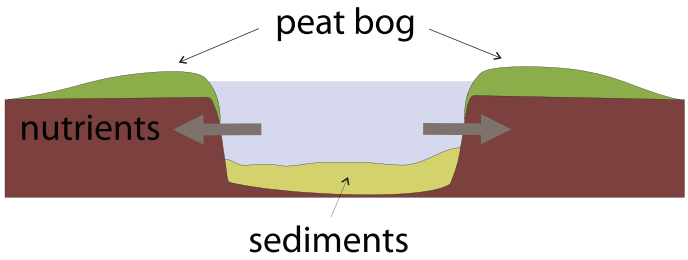


Fig. 3. Crossection of anastomosing river channel

equally supplied by river systems. As a result the bottom of valley raises covered by thick (several meters) layer of peat-bog.
Anastomosing rivers create very complex pattern of forking and joining channels. Fig. 4 presents part of real network of river Narew located in eastern Poland (Central Europe). It is relatively small anastomosing river comparing to huge anastomosing systems created by river Ob in Syberia (anastomising system can be observed from space) and Okawango in Africa.

2.2 Vascular system and tumour-induced angiogenesis

Angiogenesis is the process of formation of vascular network Carmeliet (2005). Physiologically this process occurs during embryogenesis. In mature organisms it often accompanies to cancer development and this pathological process is called tumour-induced angiogenesis. Inhibition of tumour induced angiogenesis is one of the treatments applied in oncology.
Tumour growth is a multi-stage and multi-scale process that starts with the loss of control of cell proliferation Carmeliet (2005). Initially, tumour is in avascular state and draws nutrients from surrounding vessels. At this stage cluster of tumour cells can reach size aproximately 1 - 3 mm. Tumour remains in steady state with balance between dying and proliferating cells. Without blood vessels, tumours cannot grow beyond a critical size and invade other regions of body. Tumour induced angiogenesis starts when the production of pro-angiogenic factors overcomes other forces that kept the angiogenesis quiescent so far.
Oxygen and nutrients penetrate the tissue only in a certain distance from the vessel. Distant cells, influenced by metabolic stresses, synthesise several angiogenic stimulators including VEGF (Vascular Endothelial Growth Factor) and PDGF-BB (Platelet-derived growth factor), Ang2 and NOS (Nitric Oxide Synthase) Carmeliet (2005); Steve et al. (2004). Stimulators migrate towards the nearest blood vessels. When they reach vessel, the endothelial cells (ECs) that lines the wall of this vessel are activated. They start to proliferate and migrate towards the tumour cells attracted by VEGF and other stimulators. The wall of the parent blood vessel becomes degraded and it opens to a new capillary. Migrating and proliferating ECs form a hollow tube-like cavity (the lumen), which are stabilised later by smooth muscle cells and perycites.
There are many differences between normal and tumour induced angiogenesis. Temporal and spatial expression of angiogenic factors are not well coordinated, what follows to non-uniform vascular development. As a result, new vessels form a highly chaotic and disorganised network Tonini et al. (2003). Tumour vessels are deprived various protective mechanisms i.e. perivascular cells that protect vessels from changes in oxygen or hormonal balance. Moreover the lumen does not have proper construction. Their walls may be partially constructed from cancer cells. They have also pathological form, e.g. they are thin and permeable, their diameter

Fig. 4. Pattern of river channels in anastomosing part of Polish river Narew, from Topa et al. (2006)

changes abruptly. Proliferating cancer cells can also crash the vessels. All it makes that newly formed capillaries are still unable to supply starving tumour cells with oxygen and nutrients. Inhibition of tumour-induced angiogenesis is one of the most promising strategy in anti–cancer therapy Carmeliet (2005); Ferrara & Kerbel (2005). Tumour cells deprived of nutrients and oxygen undergo necrosis. Moreover, the lack of vascular system prevents cancer cells from invading to other tissues (there is no routes, which tumour cells can use to transfer to other part of body). However, clinical tests show that none of the tested inhibitors succeed in broad range types of cancers Carmeliet (2005). Monotherapies fail because angiogenesis is controlled by very complex balance of stimulators and inhibitors. Therefore, further investigations have to concentrate on studies including wider range of angiogenic factors.

3. The models

The Graph of Cellular Automata modelling tool was initially invented for modelling anastomosing river systems Topa & Paszkowski (2001). Pure Cellular Automata model appeared too computationally demanding and it could not reflect global evolution of the

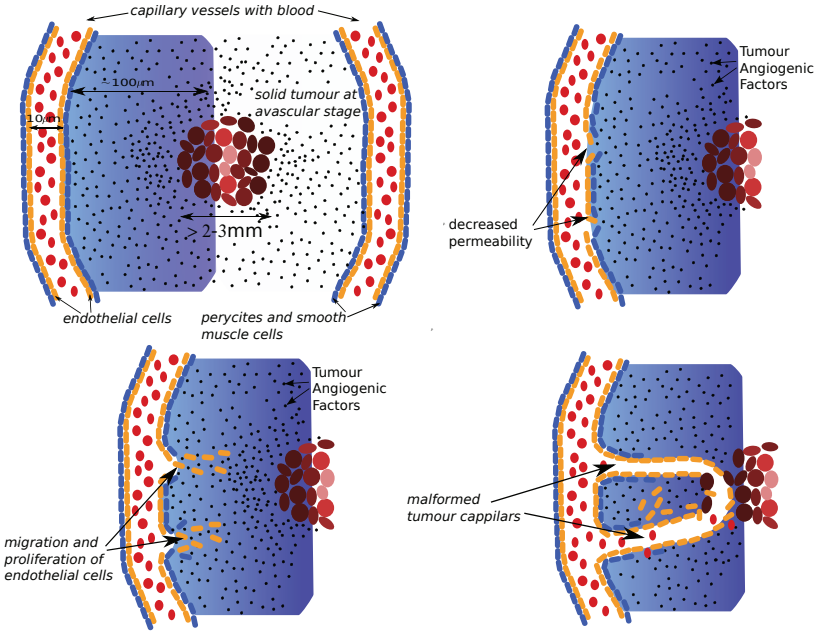


Fig. 5. Stages of tumour induced angiogenesis

system Topa et al. (2006). The model called MANGraCA (Model of Anastomosing Network with Graph of Cellular Automata) is constructed in the following way:

Cellular Automata is applied to model peat bog in the river valley and the following processes are reflected in this part of the model:

- nutrient distribution,
- growth of peat layer.

Graph of Cellular Automata is applied to model network of river channels. The following processes are implemented in this module:

- flow in river channels,
- decreasing of channel throughputs caused by peat bog development and sedimentation,
- branching, routing of new channels, anastomosing.

Cells that belong to the graph are the source of nutrients from where they are distributed over the regular mesh of automata that represent peat bog.

In more formal way the model can be defined as follows:

$$CA_{MANGraCA} = \langle Z^2, X_K, G_{CA}, S, \delta_K, \delta_G \rangle \tag{1}$$

- Z^2 — a collection of cells ordered as a square or hexadecimal mesh of $Z \times Z$ cells,
- X_K — neighbourhood for the (i, j) cell in the regular mesh of automata (Moore neighbourhood is used in the model),
- G_{CA} — a planar, directed and acyclic graph defined as (V_G, E_G), where $V_G \subset Z^2$ and $E_G \subset Z^2 \times Z^2$ are finite sets of vertices and edges, respectively,

- S — is the set of state vectors corresponding to each cell: $S = S_m \times S_g$:
  1. states for cells in regular mesh $S_m = (g, n, p)$:
     (a) g — initial altitude,
     (b) p — thickness of peat layer,
     (c) n — nutrients density,
  2. states for graph cells $S_g = (f, r)$:
     (a) f — flow rate,
     (b) r — throughput;
- $\delta_K : (g^t, n^t, p^t) \rightarrow (g^{t+1}, n^{t+1}, p^{t+1})$ — set of rules applied synchronously for each cell in Z^2.
- $\delta_G : (f^t, r^t) \rightarrow (f^{t+1}, r^{t+1})$ — set of rules applied for each cell that belong to graph.

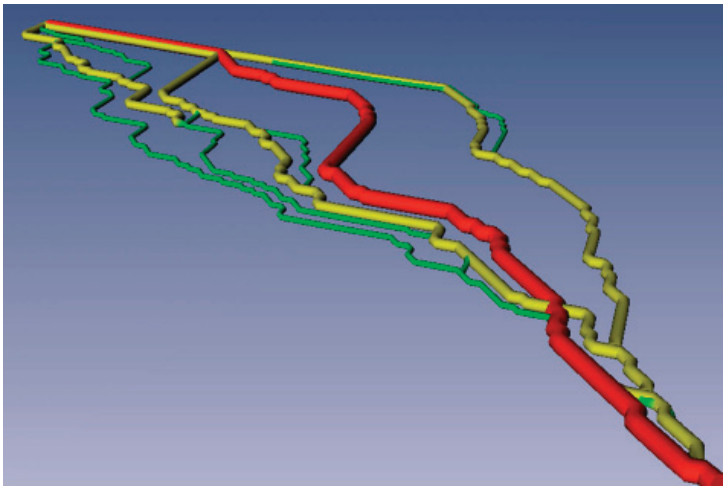


Fig. 6. Hierarchical structures of anastomosing pattern generated with model of anastomosing river.

The simulation program consists of two modules:

1. Cellular Automata Module — it calculates and update cells in regular mesh of cellular automata. It simultaneously apply defined rules of local interactions (δ_K) to each cell in the mesh and calculated new states.
   - Nutrients density (n_{ij}) is calculated as average density in neighbourhood. Cells that belong to graph have maximum density.
   - Thickness of peat layer (p_{ij}) is increased proportionally to the nutrients density.
2. Graph of Cellular Automata Module — it updates states of the cells that belong to the graph.
   - Throughput r_{ij} is decreased proportionally to thickness of peat bog layer (p_{ij}) in neighbourhood with random modifier.
   - Flow rate f_{ij} is updated asynchronously starting from "root" node of the network. The flow in river channels was calculated by using simple algorithm: the amount of water that enters the node is equal to amount that leaves node.

- Channel forming — the algorithm looks for cells with flow rate f_{ij} less than throughput r_{ij}. Such the cells are a splitting point. The route of new channel is calculated based on altitude (initial altitude and thickness of peat layer).

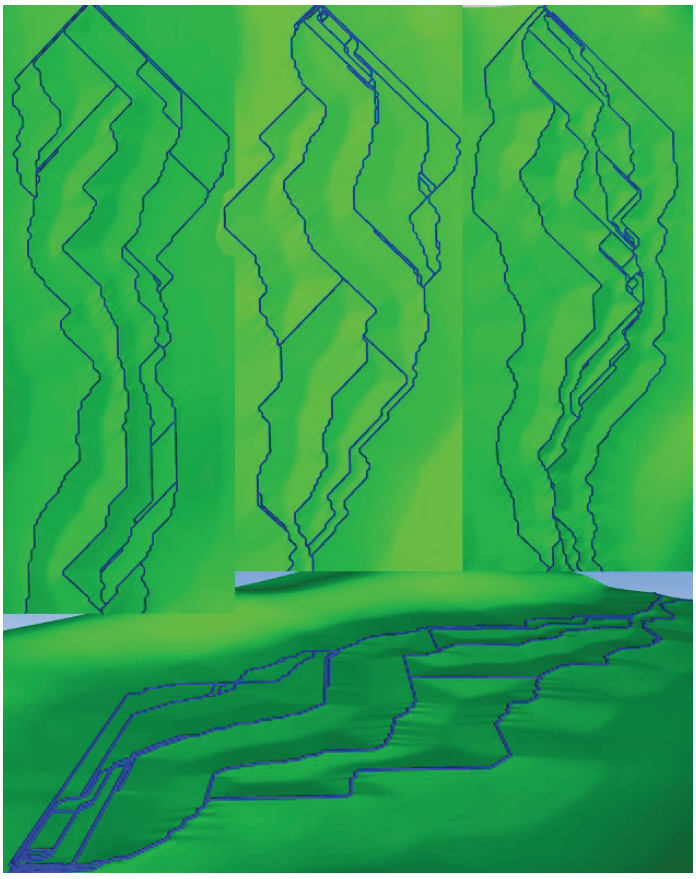


Fig. 7. Sample results of simulation anastomosing river with Graph of Cellular Automata paradigm.

The Graph of Cellular Automata was applied for modelling tumour induced angiogenesis with almost no changes in general structure Topa (2006; 2008); Topa & Dzwinel (2009).

Cellular Automata model the tissue. Some cells are initially set to represent cancer cells. In this model cancer development processes were not included. Cancer cells do not multiply during the simulation. The cancer cells are sources of various tumour angiogenic factors (e.g. VEGF, PDGF-B, Angiopoeitin) that are distributed over the mesh.

Graph of Cellular Automata is constructed over the regular mesh of cellular automata and represents vascular systems. Cells that belong to graph are the source of nutrients and oxygen in regular the mesh of automata. These two substances are spread in mesh to the neighbouring cells. The cells that are nourished sufficiently stop producing tumour angiogenic factors. Graph that represents vascular systems develops in two ways:

- growth at “tip“ cells — cells in the graph are marked as ”tip” when they are located at the end of sprout. During simulation algorithms checks the tip cells and if necessary conditions are fulfilled next neighbouring cells are added to the graph. The new cell are now the “tip” cell.
- branching — when the necessary conditions are fulfilled, cell that belong to graph can create branch (it cannot be a "tip" cell). The algorithm searches in its neighbourhood for cell that are added to graph and marked as “tip” cells for this branch. The development of the sprout proceed independently from parent vessels.

When growing vessel meets the other they join creating anastomosis.

Newly created vessels have to mature what means covering by perycites and smooth muscle cells. Only mature vessels can transport blood, supply nutrient to surrounding tissue and branch.

The model can be defined in a similar way as MANGraCA:

$$CA_{ANG} =< Z^n, G_{CA}, X_K, S, \delta > \quad \text{where :}$$

- Z^n — a collection of cells ordered as a square, hexadecimal ($n = 2$) or cubical ($n = 3$) mesh. ,
- G_{CA} — directed and acyclic graph defined as (V_G, E_G), where $V_G \subset Z^n$ and $E_G \subset Z^n \times Z^n$ are finite sets of vertices and edges, respectively,
- $X_K(i, j)$ — neighbourhood for the (i, j) cell in the regular mesh of automata,
- S — is the set of state vectors corresponding to each cell: $S = S_m \times S_g$,
  - S_m — represents the following states corresponding to all the cells in the regular CA mesh:
    - t_{ij} — state of a single tumour cell, refers to its level of supplies with nutrients,
    - taf_{ij} — TAFs concentration,
    - n_{ij} — nutrient (oxygen) concentration,
    - per_{ij} — perycites and smooth muscle cells concentration,
    - aaf_{ij} — AAF (anti-angiogenic factors) concentration.
  - S_g — represents states corresponding to the cells that belong to the Graph of Cellular Automata:
    - age_{ij} — ”age”, maturation level,
    - tip_{ij} — indicate ”tip” cell (boolean),
    - $pres_{ij}$ — pressure value,
    - $flow_{ij}$ — flow value;

The model is implemented in similar manner as the model of anastomosing river. It consist of two modules, one for calculation on regular mesh of automata and second that process graph structure:

Cellular Automata : some cells are marked as tumour cells and they are the sources of TAF (Tumour Angiogenic Factors). Cells that belong to graph act as sources of nutrients. The following processes are modelled in this module:

- TAF distribution,
- nutrients distribution — tumour cells that are supplied with nutrients stop producing TAF,

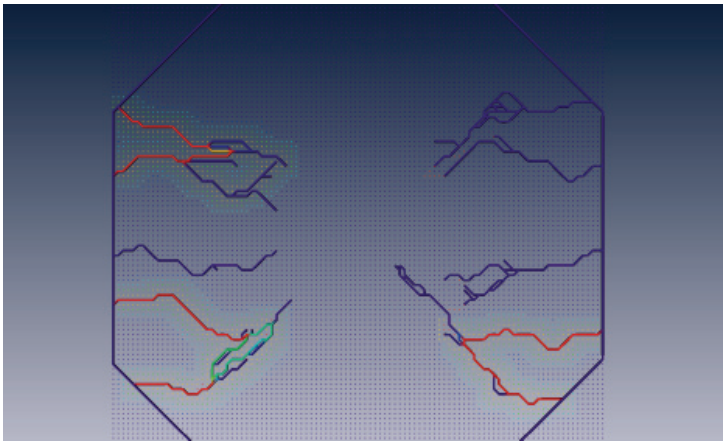


Fig. 8. Configuration with two primary vessels and cancer cells located in the center of area. Vessels are coloured according to flow value — blue means no flow, while red reflect maximum flow.

- distribution of other substances that can be included to the model (angiogenic inhibitors, perycites etc.)

Graph of Cellular Automata consists of the following processes:

- updating maturation level,
- calculating flows in vessels,
- developing sprouts — for each "tip" cells in the graph their successor is calculated,
- creating new sprouts — new sprout is created in mature cells with some probability (other conditions can be also included),

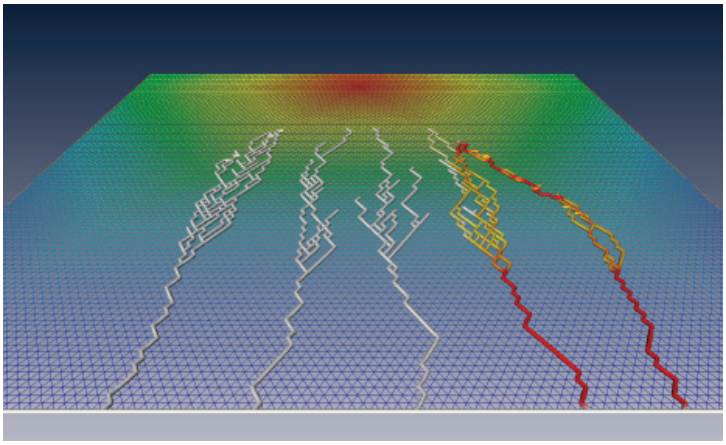


Fig. 9. Hexagonal mesh can be also applied instead of rectangular. Network segments are coloured according to flow value.

Vascular system represented by graph structure allows for convenient blood flow calculation. The model uses Poiseuille equation for calculating flow in vascular segments Mcdougall et al. (2002).

$$Q_{ij} = \frac{\pi R_{ij}^4 \Delta P_{ij}}{8 \mu L_{ij}} \tag{2}$$

- R_{ij} — segment diameter,
- L_{ij} — segment length,
- μ — viscosity,
- $\Delta P_{ij} = P_i - Pj$ — pressure difference between nodes i and j.

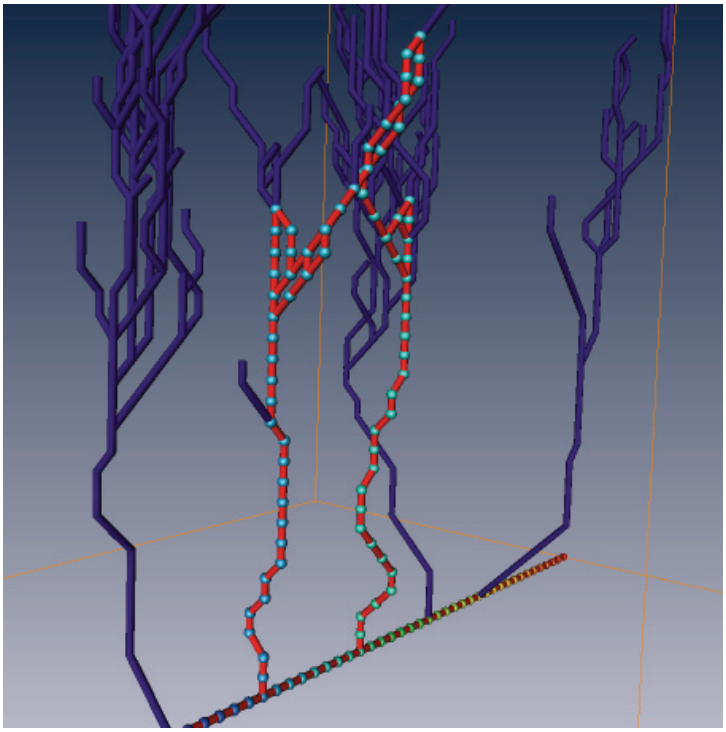


Fig. 10. Blood flow is calculated only in branches that created closed loops.

After upgrading graph structure the model calculates new flow distribution in network of vessel. It consist of three steps:

1. searching for closed loops in graph — blind sprouts are not take into consideration,
2. setting and solving system of equation for new pressures distribution in graph nodes,
3. calculating flows in graph according to pressures distribution

The framework are very flexible and we can use in the model 3D mesh (see Fig. 12). Hexagonal mesh can be also applied instead of rectangular (see Fig. 9). The model can be easily extended by new factors and processes. Figure 12 demonstrate configuration with source of angiogenic inhibitors which block vessel forming .

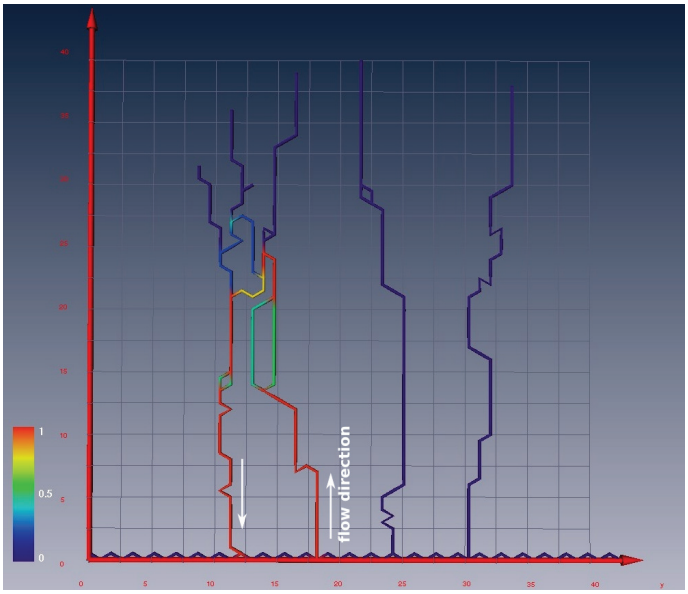


Fig. 11. Blood flow calculated in model of tumour induced angiogenesis. Vessels are coloured according to flow rate.

4. Conclusions

What is now what we called Cellular Automata? Initially Cellular Automata was treated rather as interesting mathematical model of complexity than tool for simulating complex physical phenomena. Their complex behaviour was the area of intensive and valuable studies on deterministic chaos and complexity. However when we try to reflect some real phenomena that classical definition appeared to be too limited. Some modifications were introduced to better encode modelled phenomena in Cellular Automata virtual "physics". Over time, modifications were becoming more serious. Is it still Cellular Automata? Let us enumerate what is in common in most models that use name "Cellular Automata":

- mesh, usually regular,
- evolution governed by rules,
- local neighbourhood (sometimes range is bigger then nearest cells),

It seems that those three features still are in common for models and modeling tools that use attribute "Cellular Automata".
Sometimes Cellular Automata models become very close to Agent-Based Models (ABM) approach (sometimes referred as multi-agent systems). However, it should be noticed that agents are characterized by much more complex and sophisticated behaviour of individual entities.
It is worth to emphasize that Cellular Automata paradigm its popularity as computer simulations and modelling tool partially owes the fact that it fits very well to computer implementation:

1. rules can be easily encoded in programming languages,
2. regular mesh of cells can be easily encoded in most typical data structures,

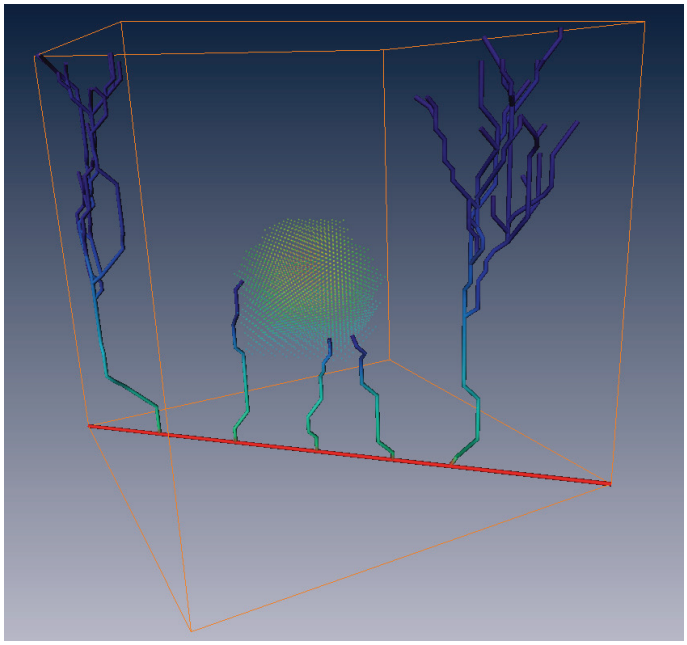


Fig. 12. Tumour induced angiogenesis in 3D. Cancer cells are located at the upper side of the box. Additionally sources of angiogenic inhibitors were located in the center of the box.

3. mesh and local neighbourhood eliminates laborious necessity of determining which elements interacts each other.
4. there is no round-off errors, numerical instabilities and so on...

In this chapter, the application of the Cellular Automata paradigm for modelling network systems was presented. The combination of Cellular Automata and graph structure was successfully applied for simulating phenomena that belong to general class of network systems located in consuming or producing environment. The examples show how broad meaning has Cellular Automata now. It should be also encouragement for further experiments with this useful paradigm that cannot be constrained by the definitions.

5. Acknowledgements

The author thanks to Witold Dzwinel (Institute of Computer Science, AGH University of Science and Technology) for his contribution to these works. The researches were partially supported by Institute of Computer Science, AGH University of Science and Technology (project no. 11.11.120.865) and also partially supported by the Polish Ministry of Education and Science (project no. NN 519 579 338).

6. References

Baxter, G. & Behringer, R. (1991). Cellular automata models for the flow of granular materials, *Physica D: Nonlinear Phenomena* 51(1-3): 465–471.

URL: *http://www.sciencedirect.com/science/article/B6TVK-46MV07J-19/2/058ea7eab54b1774410ee951bf6e69e9*

Blue, V., Embrechts, M. & Adler, J. (1997). Cellular automata modeling of pedestrian movements, Vol. 3, pp. 2320–2323 vol.3.

Cannataro, M., Gregorio, S. D., Rongo, R., Spataro, W., Spezzano, G. & Talia, D. (1995). A parallel cellular automata environment on multicomputers for computational science, *Parallel Computing* 21(5): 803–823.

Carmeliet, P. (2005). Angiogenesis in life, disease and medicine, *Nature* 438(7070): 932–936. 10.1038/nature04478.

Chopard, B. & Droz, M. (1998). *Cellular Automata Modeling of Physical Systems*, Alea-Saclay Monographs and Textes in Statistical Physics, Cambridge University Press.

D'Ambrosio, D., Gregorio, S. D., Gabriele, S. & Gaudio, R. (2001). A cellular automata model for soil erosion by water, *Physics and Chemistry of the Earth, Part B: Hydrology, Oceans and Atmosphere* 26(1): 33–39.
URL: *http://www.sciencedirect.com/science/article/B6VPV-45KWJ8F-8/2/e17522d634e01ce1769f19d5fe1cf975*

Ferrara, N. & Kerbel, R. S. (2005). Angiogenesis as a therapeutic target, *Nature* 438(7070): 967–974. 10.1038/nature04483.

Frisch, U., Hasslacher, B. & Pomeau, Y. (1986). Lattice-gas automata for the navier-stokes equation, *Phys. Rev. Lett.* 56(14): 1505–1508.

Gobron, S., Altekin, A., Bonafos, H. & Thalmann, D. (2010). Gpgpu computation and visualization of three-dimensional cellular automata, *The Visual Computer* 27: 67–81.

Gradzinski, R., Baryla, J., Doktor, M., Gmur, D., Gradzinski, M., Kedzior, A., Paszkowski, M., Soja, R., Zielinski, T. & Zurek, S. (2003). Vegetation-controlled modern anastomosing system of the upper narew river (ne poland) and its sediments, *Sedimentary Geology* 157(3-4): 253–276.

Graner, F. & Glazier, J. A. (1992). Simulation of biological cell sorting using a two-dimensional extended Potts model, *Physical Review Letters* 69: 2013–2016.

Hardy, J., de Pazzis, O. & Pomeau, Y. (1976). Molecular dynamics of a classical lattice gas: Transport properties and time correlation functions, *Phys. Rev. A* 13(5): 1949–1961.

Kier, L. B. (2000). A cellular automata model of bond interactions among molecules, *Journal of Chemical Information and Computer Sciences* 40(5): 1285–1288.

Kier, L. B., Cheng, C.-K., Tute, M. & Seybold, P. G. (1998). A cellular automata model of acid dissociation, *Journal of Chemical Information and Computer Sciences* 38(2): 271–275.

Margolus, N., Toffoli, T. & Vichniac, G. (1986). Cellular-automata supercomputers for fluid-dynamics modeling, *Phys. Rev. Lett.* 56(16): 1694–1696.

Markus, M., Bohm, D. & Schmick, M. (1999). Simulation of vessel morphogenesis using cellular automata, *Mathematical Biosciences* 156(1-2): 191–206.
URL: *http://www.sciencedirect.com/science/article/B6VHX-3W374Y2-9/2/d03620d18d402fe799635911fa3dab5c*

Masselot, A. & Chopard, B. (1996). Cellular automata modeling of snow transport by wind, *in* J. Dongarra, K. Madsen & J. Wasniewski (eds), *Applied Parallel Computing Computations in Physics, Chemistry and Engineering Science*, Vol. 1041 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 429–435.

Mcdougall, S. R., Anderson, A. R. A., Chaplain, M. A. J. & Sherratt, J. A. (2002). Mathematical modelling of flow through vascular networks: Implications for tumour-induced angiogenesis and chemotherapy strategies, *Bulletin of Mathematical Biology* 64(42): 673–702.

Neumann, J. V. (1966). *Theory of Self-Reproducing Automata*, University of Illinois Press, Champaign, IL, USA.

Reis, E., Santos, L. & Pinho, S. (2009). A cellular automata model for avascular solid tumor growth under the effect of therapy, *Physica A: Statistical Mechanics and its Applications* 388(7): 1303–1314.
URL: *http://www.sciencedirect.com/science/article/B6TVG-4V2NP1G-4/2/d7ec3cee53804a2bfa771c79ba7875bf*

Somers, J. & Rem, P. (1989). A parallel cellular automata implementation on a transputer network for the simulation of small scale fluid flow experiments, *in* G. van Zee & J. van de Vorst (eds), *Parallel Computing 1988*, Vol. 384 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 116–126.

Steve, N. M., Webb, S. & Othmer, H. G. (2004). Mathematical modeling of tumor-induced angiogenesis, *J. Math. Biol* 49: 111–187.

Toffoli, T. & Margolus, N. (1987). *Cellular automata machines: a new environment for modeling*, MIT Press, Cambridge, MA, USA.

Tonini, T., Rossi, F. & Claudio, P. P. (n.d.). Molecular basis of angiogenesis and cancer, *Oncogene* 22(42): 6549–6556.

Topa, P. (2006). Towards a two-scale cellular automata model of tumour-induced angiogenesis, *in* S. E. Yacoubi, B. Chopard & S. Bandini (eds), *ACRI*, Vol. 4173 of *Lecture Notes in Computer Science*, Springer, pp. 337–346.

Topa, P. (2008). Dynamically reorganising vascular networks modelled using cellular automata approach, *Cellular Automata, 8th International Conference on Cellular Automata for Reseach and Industry, ACRI 2008, Yokohama, Japan, September 23-26, 2008. Proceedings*, Vol. 5191 of *Lecture Notes in Computer Science*, Springer, pp. 494–499.

Topa, P. & Dzwinel, W. (2003). Consuming environment with transportation network modelled using graph of cellular automata, *in* R. Wyrzykowski, J. Dongarra, M. Paprzycki & J. Wasniewski (eds), *PPAM*, Vol. 3019 of *Lecture Notes in Computer Science*, Springer, pp. 513–520.

Topa, P. & Dzwinel, W. (2009). Using network descriptors for comparison of vascular systems created by tumour-induced angiogenesis, *Theoretical and Applied Informatics* 21(2): 83–94.

Topa, P., Dzwinel, W. & Yuen, D. A. (2006). a Multiscale Cellular Automata Model for Simulating Complex Transportation Systems, *International Journal of Modern Physics C* 17: 1437–1459.

Topa, P. & Paszkowski, M. (2001). Anastomosing transportation networks, *in* R. Wyrzykowski, J. Dongarra, M. Paprzycki & J. Wasniewski (eds), *PPAM*, Vol. 2328 of *Lecture Notes in Computer Science*, Springer, pp. 904–912.

Wang, H., Nie, G. & Fu, K. (2008). Cellular automata simulation of the growth of bone tissue, *ICNC '08: Proceedings of the 2008 Fourth International Conference on Natural Computation*, IEEE Computer Society, Washington, DC, USA, pp. 421–424.

Was, J. (2005). Cellular automata model of pedestrian dynamics for normal and evacuation conditions, *ISDA '05: Proceedings of the 5th International Conference on Intelligent Systems Design and Applications*, IEEE Computer Society, Washington, DC, USA, pp. 154–159.

Wolfram, S. (2002). *A New Kind of Science*, Wolfram Media.
URL: *http://www.amazon.com/exec/obidos/ASIN/1579550088/ref=nosim/rds-20*

Wooldridge, M. (2009). *An Introduction to Multiagent Systems*, Wiley.

13

Cellular Automata Modeling of Biomolecular Networks

Danail Bonchev
Virginia Commonwealth University
USA

1. Introduction

On the eve of the new, 21st century a paradigm shift began in biology and biomedical research. After decades of meticulous studies of individual genes and proteins, and their biological functions, time was ripe for the contours of the forest to start emerging behind the trees. Failures with some new drugs showing unsuspected harmful side effects, along with similar cases in gene engineering, have signaled that the old reductionist approach has its limits. It has been overoptimistic to expect to cure a sickness by curing a single defective gene or a single incapacitated protein, because genes and proteins do not exist and act in isolation; they are part of a system. The new systemic approach in biology and medicine requires to account for the environment in which biomolecules act within the living cell and intercellular space. This environment is organized in complexes, pathways, and networks, containing hundreds and thousands of biomolecules. The essence of the new science of *systems biology* (Kitano, 2002; Ideker, 2004; Alon, 2006; Palsson, 2006; Choi, 2007) and *systems medicine* (Nadeau & Subramanian, 2010) had to be expressed in the language of networks, which are the best means of defining a system as a whole and explaining its features and functions.

Could this postgenomic era start earlier? The answer is: "*Yes* and *No*". *Yes*, because facts for the limitations and pitfalls of the reigning paradigm have been accumulating for a long time, although genomics had still to wait to reach its peak with the advent of the new sequencing technologies and the flood of genetic data that followed. *No*, because the theoretical foundation and the computational tools were still lacking. Network theory was known for a century and a half since the theory of electrical systems has been proposed by Kirchhoff in 1845. Kirchhoff's work is considered as one of the three pillars of graph theory, along with the Oiler'famous Königsberg bridges problem (1736), and the problem with calculating the number of isomeric compounds in chemistry, investigated first by Cayley in 1874. The second part of the 20th century in graph theory has been marked by the great authority of Erdös (Hoffman, 1998) and Bollobás (2001), which also developed the basics of the theory of random networks. Unfortunately, working within the framework of pure mathematics, these brilliant mathematicians have not been interested in the complex dynamic networks of the real world. There has been a considerable development of theory in social networks (Scott, 1987; Borgatti et al., 2009), however, the point of no return in network theory was reached only at the end of the 1990s (Neuman et al., 2006).

Watts & Strogatz established an important property of complex nonrandom networks - their small diameter - and termed such networks "small-world" ones (Watts & Strogatz, 1998,

Strogatz, 2002, Watts, 1999, 2003). The meaning of this finding is that genes and proteins in the living cell are only few steps away; they are much more strongly intertwined than previously supposed. It was soon confirmed for almost any type of complex networks that they share this property of *smallworldness* (Neuman, 2003). A major contribution of Barabási and coworkers (Barabási & Albert, 1999; Albert et al., 2000; Albert & Barabási, 2000; Barabási, 2002; Barabási **&** Oltvaj, 2004) summarized other common properties of these networks. It was shown that the node degrees in them are distributed in a specific way characterized with a presence of a few highly connected nodes, whereas the great majority of nodes are of low degree. As a whole the degree distribution is *scale*-free, and follows *a power law* with a negative exponent within the -2.0 to -2.5 range. The highly connected nodes, called *hubs*, were found to play important role in network stability (resilience against random attacks), while on the negative side being also responsible for spreading attacks directed to them through the network in events like epidemics in social networks, vulnerability of ecosystems, etc. The existence of this specific degree distribution in complex networks of different nature was derived from models of network evolution in which new nodes are preferentially attached to nodes with high degrees. Later work (Dorogovtsev et al., 2000; Dorogovtsev & Mendes, 2001) has shown that laws other than the power law could also take place in complex networks, and other patterns of network evolution also play important role.

The specificity of the complex dynamic networks was also extended to their overall *modular structure* (Rives & Galitski, 2003**;** Neumann & Girwan, 2004, Guimera & Amaral, 2005; Newman, 2006) and their local topology as characterized by high degree of *clustering* (Friedkin, 1984, 1990;) and specific network *motifs* (R. Milo et al., 2002; Wernicke & Rasche, 2006; Alon, 2007). Modularity is also called network's *community structure*. A high degree of modularity implies high degree of connectivity within the modules, while considerably less degree of intermodular connectivity (Reichardt & Bornholdt, 2006). Clustering coefficient measures the degree to which nodes in a network tend to cluster together. In complex dynamic networks, this likelihood tends to be considerably greater than that in random networks of the same size and the same node degree distribution (Watts & Strogatz, 1998; Barrat, 2004; Opsahl, T. & Panzarasa, 2009). Network motifs are subgraphs that occur in real-world networks more frequently than expected in random graphs of comparable size and connectivity. Different types of networks are characterized by their specific motif signature - a preferred small set of subgraphs. The question of whether the motif signature is related to function is still a subject of controversy (Knabe et al., 2008; Konagurthu & Lesk, 2008).

Despite of the young age, the network analysis of complex systems has demonstrated its capacity to produce valuable information in the fields of molecular biology and medicine. Patterns of evolution have been captured studying the evolution of network structure and complexity (Weitz et al., 2007; Hinze & Adami, 2008; Knabe et al., 2008; Mazurie et al., 2010). The detailed characterization of network structure by topological and information-theoretic descriptors provided means for successful phylogenetic reconstruction (Mazurie et al., 2008). The networks of gene, protein and metabolic interactions of model organisms like yeast, fruit fly, and the nematode *C. elegans*, became invaluable resource for modeling human biology, pathology and longevity (Managbanag et al., 2008), and helped in identifying protein markers for cancer and other diseases. The building of the human protein-protein interaction network (the unfinished yet Human Proteome Organization project (HUPO, 2002)) has already help to trace down the effect of drugs on different molecular pathways, raising the hopes for improved drug discovery methods (Butcher et al.,

2004; Hopkins, 2008). All these endeavours have been greatly helped by high level professional software tools (Thomas & Bonchev, 2010) like Ingenuity Pathway Analysis (Ingenuity Systems), Pathway Studio (Ariadnegenomics), Cytoscape (The Cytoscape Software), Fanmod (Wernicke & Rasche), and others, along with publicly available databases for all kinds of biomolecular interactions (KEGG, http://www.genome.jp/kegg/kegg1.html; PINA, http://csbi.ltdk.helsinki.fi/pina/; Gene Ontology, http://www.geneontology.org/GO.databa se.shtml).

Yet, this explosive development of network theory concerns mainly network structure, rather than network dynamics; networks are static. Many of molecular biology level networks, like protein-protein interactions ones, incorporate all possible interactions, but not only those which are active at a given moment in time. Dynamics of the processes down the numerous network pathways remains largely untouched. The modeling of this dynamics by differential equations (ODE) marked certain success in several specific intracellular processes. The regulation of cell cycle (the sequence of steps by which a cell replicates its genome and distributes the copies between the two daughter cells) received a considerable attention (Tyson, 2001, Csikasz-Nagy et al., 2006). Another series of elaborate models has been focused on regulation in network motifs (the small building blocks of networks, containing several nodes), (Milo et al., 2002) in gene regulatory networks (Mangan & Alon, 2003; Alon, 2006, 2007; Longabaugh & Bolouri, 2006). The high complexity of real-life networks and the lack of experimental kinetic data make constructing of this type of models impractical not only computationally, but even at the stage of defining the very set of equations.

Related to the above mentioned, the aim of this chapter is to show that cellular automata (CA) modeling technique could partially fill the gap in describing the dynamics of biomolecular networks. While not able to provide exact quantitative results, it will be shown that the CA models capture essential dynamic patterns, which can be used to control the dynamics of networks and pathways. CA models of human diseases can help in the fight against cancer and HIV by simulating different strategies of this fight. Another field of application presented is the performance rate of network motifs with different topology, which might have evolutionary and biomedical importance.

2. Cellular automata

2.1 Previous work on CA models of biological systems

The early attempts to model biological systems by cellular automata (CA) have included developmental biology, population biology and neurobiology, along with blast aggregation, neuronal maps, and branching networks, as well as several classical cases of pattern formation (Ermentrout & Edelstein-Keshet, 1993). Quantitative spatial and temporal correlations in sequences of chlorophyll fluorescence images from leaves of *Xanthium strumarium* have been reproduced by cellular automata models with a high statistical significance (Peak et al., 2004). Dynamics of biological networks was investigated by Kauffman who proposed models of random genetic regulatory networks (Kauffman 1969, 1993). These discrete random Boolean networks (RBNs) are named after him as Kauffman (or *NK*) networks. The models have been used as a basis for the concept of self-organization and emergence of life from randomness, viewing life as a state intermediate between chaos and complete order (Kauffmann, 1993). A step toward more realistic models of Boolean dynamics of biological networks has been to use random networks with scale-free topology

(Aldana, 2003; Kauffman, 2003). The dynamical property of stability or robustness to small perturbations has been found to correlate highly with the relative abundance of specific network motifs in several biological networks (Prill et al., 2005). Such findings support the views for system dynamics strong dependence on network structure.
Networks of biomolecules in the living cell have most frequently elementary steps of enzymatic chemical reactions. The first CA model of an enzymatic reaction has been proposed in 1996 (Kier et al., 1996) and, being prematurely born, remained unnoticed for some time. With the "phase transition" in network theory from random to complex real-life network such a CA approach to “enzymatic reactions networks" was independently proposed in the beginning of the new century (Weimar, 2002). These ideas were developed extensively in the following years in the Center for the Study of Biological Complexity at VCU in Richmond, Virginia (Kier & Witten, 2005; Kier et al., 2005; Bonchev et al., 2006; 2010; Apte et al., 2008, 2010; Taylor et al., 2010).

2.2 The Cellular automata method as applied to network dynamics analysis

Cellular automata (CA) are mathematical machines, which describe the behavior of discrete systems in space, time, and state. CA are a powerful modeling technique with a broad field of applications including mathematics, chemistry, physics, biology, complexity and systems science, computer sciences, social sciences, etc. It has been developed by the mathematical physicist John von Neumann in the mid 1940s, in collaboration with Stanislaw Ulam (von Neumann, 1966). Their pioneering work on self-reproducing automata opened the door to the fascinating area of artificial life. The method became popular in the 1970s with the "Game of Life" of John Conway, popularized by Martin Gardner in Scientific American (Gardner, 1970). A general theory of cellular automata as models of the complex world was proposed by Steven Wolfram, who later advocated cellular automata as an alternative way of making science, an approach that can reproduce not only the known scientific truths, but also open the door to new discoveries (Wolfram, 1986; 2002). A further generalization of the simple CA rules that produce complex behavior was offered by Rücker in his theory of the universal automatism (Rücker, 2005).
Cellular automata have five fundamental features (von Neumann, 1966):

1. They consist of a discrete lattice of cells (1D, 2D or 3D).
2. They evolve in discrete time steps (iterations), beginning with an initial state at time t = 0.
3. Each site takes on a finite number of possible values, the simplest being "occupied" and "unoccupied".
4. The value of each site evolves according to the same rules (deterministic or probabilistic ones).
5. The rules for the evolution of a site depend only on the local *neighborhood* of sites around it.

Each cell in the most commonly used square lattice has four neighbor sites (von Neumann neighborhood) and four extended neighbor sites located next to the cell corners (extended von Neumann neighborhood). To avoid "edge effects", the lattice is usually embedded on the surface of a torus. The cell is the basic model of each of the system elements. Its state may change at the next iteration. The contents of a cell may either break away or move to join an occupied neighboring cell. The question which movement will be chosen depends upon the modeled system. The movement of the cells may be simultaneous (synchronous), or the rules may be applied to each cell at random, until all cells have computed their states and trajectories (asynchronous movement). This constitutes one iteration, a unit of time in

the cellular automata simulation. The initial state of the system is random and, thus, does not determine subsequent configurations at any iteration. The same set of rules does not yield the same configurations, except in average. The configurations after many iterations reach a collective organization that possesses relative constancy in appearance and in reportable counts of cells. These are the emergent characteristics of a complex system.

In simulating enzymatic reactions organized in a network it usually suffices to use a 2D-square lattice, with cells partially occupied by molecules and controlled by several simple rules. These are rules describing the probabilities of two adjacent cells to separate, to join, or to change their state after joining. The first rule defines the movement probability, P_m, as a probability that an occupant in an unbound cell will move to one of the four adjacent cells, if that space is unoccupied. If it moves to a cell whose neighbor is an occupied cell, then a bond will form between these cells. The second rule describes the probability for molecule at cell A, to join with a molecule at cell B, when an intermediate cell is vacant. The joined cells can separate again, depending on the breaking probability, P_B. When molecule A is bonded to two molecules, B and C, the simultaneous probability of a breaking away event from both B and C is $P_B(AB)*P_B(AC)$.

In this chapter we follow the general approach used by Kier and Cheng (Kier et al. 1996, 2005a, 2005b) in setting up a CA model of enzyme activity. The mechanism of the enzymatic reaction is assumed to start with an interaction between the substrate *S* and enzyme *E*, which form a *SE* complex. The latter is rearranged to a complex *PE* between the enzyme E and the product P, which are then separated and the enzyme molecule E is free to take part in another interaction:

$$S + E \rightarrow SE \rightarrow PE \rightarrow P + E \quad (1)$$

Focusing more on identifying patterns characterizing the (quasi-) steady state reached after many iterations, rather than on the temporal changes, our models are spatial ones. A network to be studied is represented by groups of CA cells, each group including one of the network species: enzymes, substrates, or products. The number of cells in each group is selected so as to reflect the relative concentrations of each network species. Each group of cells moves freely in the grid. The only cell encounters that change the CA configuration are those between a specific substrate and a specific enzyme. When such an encounter occurs, an enzyme-substrate complex is formed. The complex has an assigned probability of changing to a new complex (enzymatic product). Following this, another probability is assigned for the separation of the product from the enzyme. The movement probability, P_m, determines the extent of any movement. Thus, for an enzyme cell, $P_m = 0$ would designate a stationary enzyme. The CA model selected is asynchronous. Cells compute their states one at a time. In our study, all three types of probabilities were assumed equal to unity: $P_m = P_b = P_j = 1$. This means that all cells may interact, join, and break apart with equal probability. Only the cells involved in a specific state change, i.e., enzyme - substrate (*ES*) or enzyme - product (EP), are endowed with a state-changing probability rule, defined by the transition probability P_c, which describes the probability of an *ES* pair of cells changing to an *EP* pair of cells. It may be regarded as a measure for enzyme activity or efficiency. The collection of rules associated with a network species thus represents a profile of the structure of that species and its relationship with other species. By systematically varying the rules, one can arrive at a profile of configurations reflecting the influences of different species.

In modeling the dynamics of a signaling pathway the first goal is to show whether the model reproduces the amplification of the signal through the pathway. The next goal is to examine the pathway sensitivity to a variety of initial conditions, and to reproduce experimentally found patterns of substrate and product variations. Analyzing the findings the ultimate goal is to define the ways to control the pathway dynamics toward a desirable outcome. In what follows we present evidences that the CA method is capable of providing an answer to all these questions.

3. The EGF-induced MAPK signaling pathway as a case study for applying cellular automata to pathways and networks (Kier et al., 2005c)

Mitogen-activated protein kinase (MAPK) pathways are major signaling cascade controlling complex programs such as embryogenesis, differentiation, and cell death, in addition to short-term changes required for homeostasis and hormonal response, gene transcription and cell cycle progression. The molecular mechanism of this pathway has been studied intensively by different numerical methods (differential equations, stochastic approaches, etc.) based on reaction-rate equations (Huang & Ferell, 1996; Bhalla & Iyengar, 1999; Kholodenko, 2000, McCullagh et al., 2010). Our cellular automata modeling was limited to the major cascade part of the pathway, which has been incorporated in all biochemical models proposed so far. The cascade is shown in Figure 1. The detailed reaction mechanism of the MAPK cascade is shown below in terms of the elementary enzyme reactions:

$$A + E1 \rightarrow AE1 \rightarrow BE1 \rightarrow B + E1$$

$$B + E2 \rightarrow BE2 \rightarrow AE2 \rightarrow A + E2$$

$$C + B \rightarrow CB \rightarrow DB \rightarrow D + B$$

$$D + B \rightarrow DB \rightarrow EB \rightarrow E + B$$

$$D + E3 \rightarrow DE3 \rightarrow CE3 \rightarrow C + E3$$

$$E + E3 \rightarrow EE3 \rightarrow DE3 \rightarrow D + E3$$

$$F + E \rightarrow EF \rightarrow EG \rightarrow G + E$$

$$G + E \rightarrow EG \rightarrow EH \rightarrow H + E$$

$$G + E4 \rightarrow GE4 \rightarrow FE4 \rightarrow F + E4$$

$$H + E4 \rightarrow HE4 \rightarrow GE4 \rightarrow G + E4$$

The 2D-CA models were built from the above reaction mechanisms using a 100 x 100 grid. The probabilities of joining and breaking away cells were assumed to be equal to unity. Each of the models was obtained as the average of 50 runs, each of which included 5000 to 15000 iterations, a number sufficiently large to enable reproducing the steady state (or nearly steady state) of the set of reactions examined. The three substrates MAPKKK, MAPKK, and MAPK, and the four enzymes involved, have some prescribed initial concentrations (a number of CA cells). We have systematically altered the initial concentrations of the above

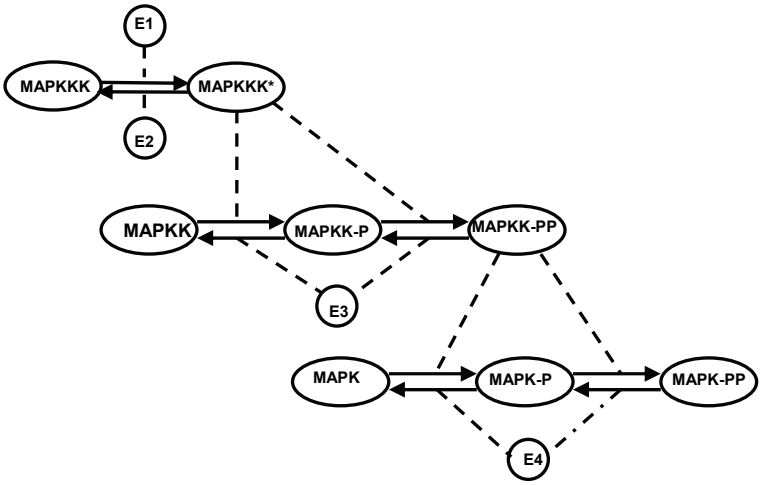


Fig. 1. The MAPK signaling cascade. The catalytic reactions of phosphorylation (*P*) and diphosphorylation (*PP*) are helped by enzymes *E*1-*E*4, as well as by the activated MAPKKK and MAPK-PP kinases. (Courtesy of "Chemistry and Biodiversity" journal (Kier et al., 2005)).

substrates, as well as the efficiencies of the enzymes. The basic variable was the initial concentration of MAPKKK, which was varied within a 25-fold range from 20 to 500 cells, matching thus the 25-fold range of variation of *E*1 used as an initial stimulus in (Huang & Ferrell, 1996). The concentrations of MAPKK and MAPK were kept constant (500 or 250 cells) in most of the models. The four enzymes, denoted by E1, E2, E3, and E4, were represented in the CA grid by 50 cells each. In one series of models, we kept the MAPKKK initial concentration equal to 50 cells, and varied the transition probabilities of one of the enzymes within the 0 to 1 range, while keeping constant (P_c = 0.1) those of the other three enzymes. In another series, *all* enzyme transition probabilities were kept constant (P_c = 0.1), whereas the concentrations of substrates were varied. A third series varied both substrate concentrations and enzyme propensities. The variations in the concentrations of all eight species (the three substrates MAPKKK, MAPKK, and MAPK, and the five products MAPKKK*, MAPKK-P, MAPKK-PP, MAPK-P, and MAPK-PP, denoted in the set of equations as A, C, F, B, D, E, G, and H, respectively) were recorded.

The simulation produced temporal plots, which express the changes in the substrates and products concentrations up to reaching a steady state. The steady-state concentrations of all species were then used to construct spatial models of concentration dependence on the enzyme propensity and other variables of the process. The enzymes activity is controlled by inhibitors, a process that is simulated by cellular automata for the entire probability range of 0 to 1. An example with the concentration profile of the MAPK cascade at variable propensity of enzyme E3 is shown in Fig. 2.

It was found that the maximum amplification of the cascade signal (the largest production of the doubly phosphorylated MAPK, denoted as species H) occurs at a narrow range of intermediate propensity of enzyme *E*3, due to the reversing of the second row phosphorylation reactions. This result confirms the expectations that the CA models can predict dynamic patterns and help in finding optimum conditions for the input signal amplification.

Better results in the search for optimal ranges of parameters can be obtained by using 3D- or contour plots. Such a plot in Fig. 3 provides optimal ranges of the initial concentration of the

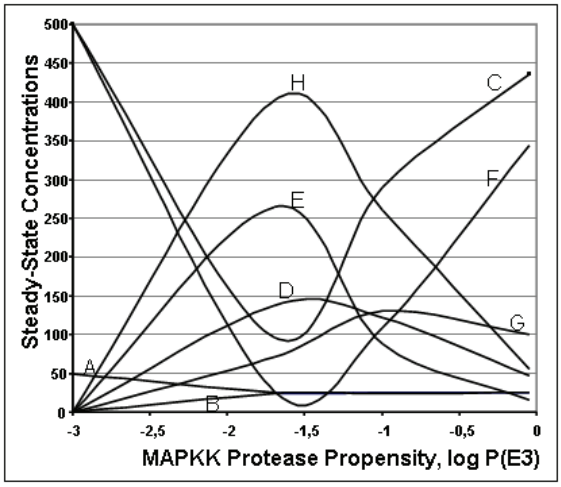


Fig. 2. A spatial model of the concentration dependence of the eight MAPK proteases on the propensity of enzyme E3. A narrow range of the enzyme propensity defines the optimal concentration of the cascade product H and the intermediate E. (Courtesy of "Chemistry and Biodiversity" journal (Kier et al., 2005)).

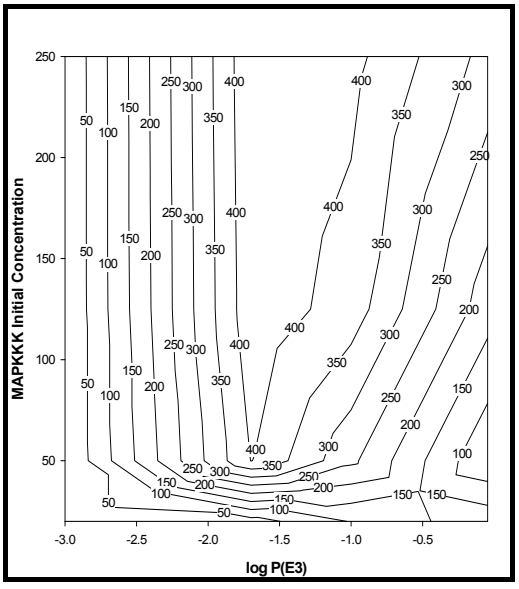


Fig. 3. A contour plot defining the optimal ranges of the MAPKKK initial concentration and the enzyme E3 activity needed to reach the maximum amplification of the cascade outgoing chemical signal MAPK-PP (the contour line of 400 cells). (Courtesy of "Chemistry and Biodiversity" journal (Kier et al., 2005)).

cascade input substrate A (MAPKKK) and the propensity of enzyme *E*3, needed for reaching a maximal amount of the cascade target product H (MAPK-PP). More specifically, the contour line with MAPK-PP concentration of 400 cells indicates that such optimal conditions can be realized with MAPKKK initial concentration of at least 50 cells and the enzyme E3 activity should be a moderate one (corresponding to the logarithmic range of -1.5 to -1.8).
An important outcome of our CA modeling of the MAPK signaling cascade is the possibility to summarize the patterns of network dynamics in a set of recommendations how to manipulate the network variables in order to achieve a certain result (Table 1). Such a method for pathway control could be of particular importance for the field of drug discovery. Searching to design can also reveal specific mechanistic details of the system studied. Such a conclusion can be drawn from Figs. 4a,b, which show a sigmoid curve of the cascade product *H* dependence on the initial concentration of the source substrate *A*. Such curves deviating from Michaelis-Menten kinetics are a characteristic fingerprint of cooperative effect of cascade enzymes. Our finding confirmed the result obtained in (Huang and Ferell, 1996) by numerically solutions of the differential rate equations.

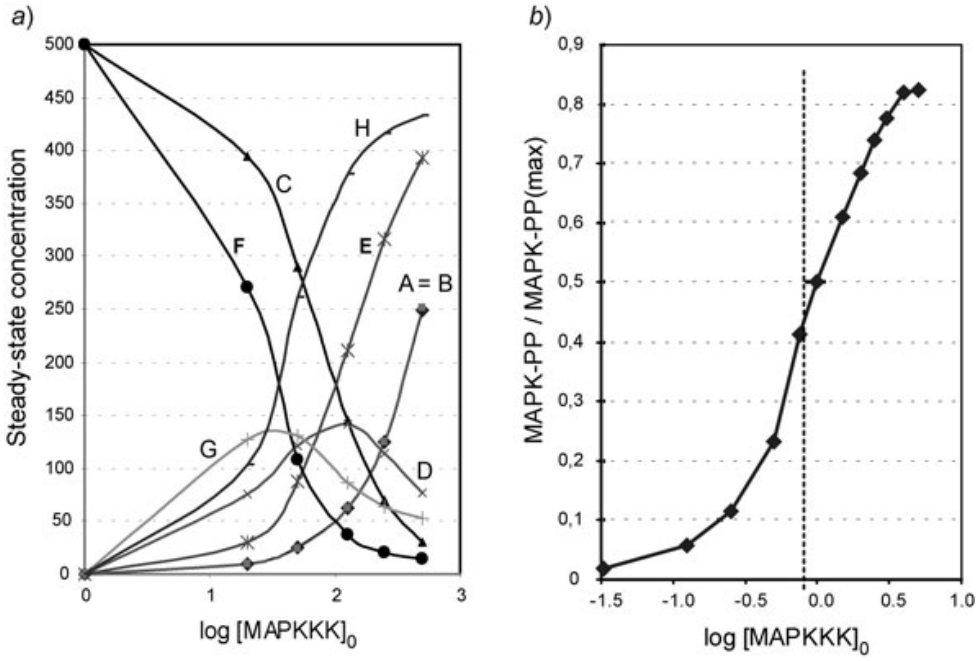


Fig. 4. a) Steady-state concentrations of substrates and products dependence on the initial concentration of MAPKKK. Model parameters used : Enzymes E1-E4 transitional probabilities equal to 0.1; initial concentrations of substrates C and F - 500 cells; b) Relative stimulus/response ($MAPKKK_0$/MAPK-PP) plot with $MAPKKK_0$ expressed in multiples of EC50. The slope of the H and MAPK-PP curves in the figures evidences for the significant cascade-signal amplification, while the S-shape of the curves confirms the hypothesis for enzymes cooperative action. (Courtesy of "Chemistry and Biodiversity" journal (Kier et al., 2005)).

Objectives		Action needed	Propensity Range
Decrease [MAPK]	→	Inhibit E2, E3, E4	P = 0.9 → P = 0.02
Increase [MAPK]	→	Inhibit E1	P = 0.9 → P = 0
Decrease [MAPK-PP]	→	Inhibit E1	P = 0.9 → P = 0
Increase [MAPK-PP]	→	Inhibit E3, E4	P = 0.9 → P = 0.02
Decrease [MAPKK]	→	Inhibit E3	P = 0.9 → P = 0.02
Increase [MAPKK]	→	Inhibit E1	P = 0.9 → P = 0

Table 1. Inhibiting enzymes E1 to E4 as a tool for controlling the MAPK pathway CA simulations

4. CA models of Apoptosis pathway as a tool for developing strategies to fight cancer

4.1. Cellular automata modeling of the FASL- Activated Apoptosis pathway

Apoptosis is a process of programmed cell death, the most common mechanism by which the body eliminates damaged or unneeded cells such that threaten the organism survival (Wajant, 2002). A number of diseases, including cancer and HIV, are associated with abnormal functioning of apoptosis (Fadeel & Orrenius 2005; Eils et al., 2009). Devising strategies for manipulating apoptosis would have a major impact on drug discovery process, which explains the considerable interest to this topic (Brajušković, 2005; Fulda & Debatin, 2004**;** Hanahan & Weinberg, 2000; Lowe et al., 2004; Marek et al., 2003; Reed, 2006).

Apoptosis can be induced by two types of signaling cascades, *intrinsic* and *extrinsic* ones, the proteins from which are of considerable interest as drug targets. The intrinsic pathways are activated by developmental signals or severe cell stress caused by different environmental factors. The extrinsic signaling is initiated by different chemical signals, such as FAS ligand (FASL). The latter binds to the death receptor FAS (CD95), which induces the formation of the death-inducing signaling complex (DISC) by attracting the FAS-associated death domain protein (FADD) and the *initiator caspases* 8 or 10 (Fig. 1). The recruitment of the two caspases is favored by the formation of a FAS homodimer and a lattice with ordered FAS-FADD pairs. The spatial proximity of CASP8 and CASP10 in the complex triggers their autocatalytic activation and their release into the cytoplasm where they activate CASP3, CASP6, and CASP7 termed *effector caspases*. The activated CASP3 and CASP7 split the heterodimer DFF (DNA Fragmentation Factor), and the released DFF40 starts the DNA fragmentation. CASP6 cleaves the caspase substrates, contributing further to the cell distraction. The pathway is regulated by c-FLIP (FADD-like apoptosis regulator) protein and the IAP (Inhibitor of APoptosis) protein family, from which XIAP is the most potent inhibitor (Salvesen et al., 2009; Scott et al., 2009).

Using cellular automata we simulated two strategies to fight cancer by modulating the FASL-induced apoptosis. The first strategy builds on recent publications elucidating important details of the role of T-cells in the immune response to fight cancerous and HIV-infected cells (Ferguson & Griffith, 2006). Tumors counterattack the immune system by inducing apoptosis in T-cells using overexpression of FASL, while preventing their own destruction by the same apoptotic mechanism (Igney & Krammer, 2005). In our study (Apte et al., 2010) we simulated a strategy to fight cancer and HIV by blocking the apoptosis in T-cells via maximizing the effect of FLIP and IAP inhibitors (Fig. 5).

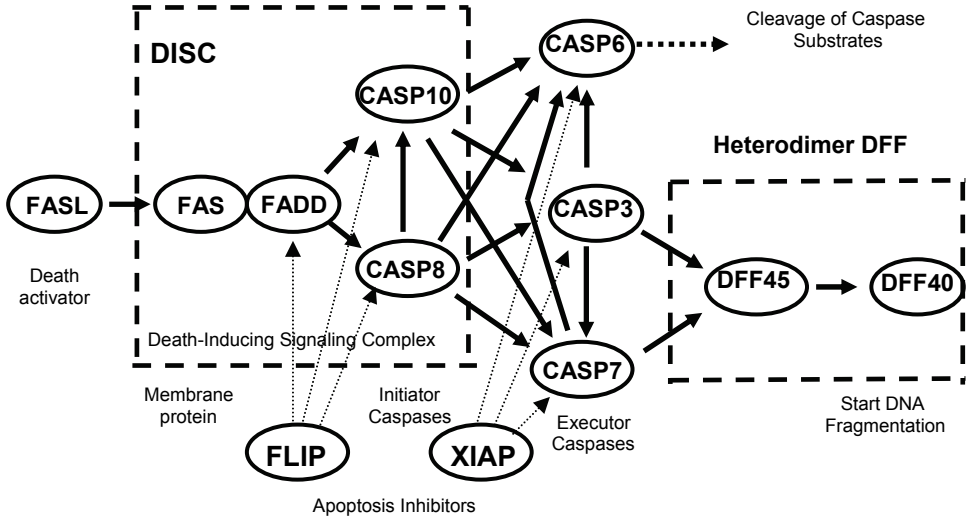


Fig. 5. The apoptosis pathway activated by the FASL protein (Bonchev et al., 2006). A cascade of activations of caspase (CASP) proteases releases the DNA Fragmentation Factor DFF40, which starts the DNA fragmentation, while CASP6 cleaves caspase substrates. The apoptosis performance can be widely modulated using inhibitors FLIP and XIAP (Wilson *et al.,* 2009; Irmler et al., 2009).
(Cellular Automata (CA) Modeling of Biomolecular Networks Dynamics, D. Bonchev, S. Thomas, A. Apte, L. B. Kier, SAR & QSAR in Environmental Research, 2010, reprinted by permission of Taylor & Francis Ltd) http://www.informaworld.com).

The detailed set of equations used as an input for the CA simulation is shown below. It matches the mechanistic information on the FASL-triggered apoptosis discussed in the foregoing. The abbreviation used read as follows: An asterisk* stands for "activated"; A-B means complex of A and B; DISC1 and DISC2 stand for the FAS/FADD/CASP8* and FAS/FADD/CASP10* complexes, respectively.

FAS + FAS-L → FAS* + FAS-L	(Ligand attachment)
FAS* + FAS* → FAS*-FAS*	(DISC recruitment)
(FAS*)2 + FADD → FAS-FADD*	(DISC recruitment)
FAS-FADD* + CASP8 → DISC1*	(DISC complex formation)
FAS-FADD* + CASP10 → DISC2*	(DISC complex formation)
DISC1* + FLIP → FLIP-DISC1	(Inhibition)
DISC2* + FLIP → FLIP-DISC2	(Inhibition)
CASP8* + CASP10 → CASP10* + Casp8*	(CASP activation)
CASP8* + CASP3 → CASP3* + CASP8*	(CASP activation)

CASP8* + CASP6 → CASP6* + CASP8* (CASP activation)

CASP8* + CASP7 → CASP7* + CASP8* (CASP activation)

CASP10 + CASP3 → CASP3* + CASP10* (CASP activation)

CASP10 + CASP6 → CASP6* + CASP10* (CASP activation)

CASP10 + CASP7 → CASP7* + CASP10* (CASP activation)

CASP3* + CASP6 → CASP6* + CASP3* (CASP activation)

CASP3* + CASP7 → CASP7* + CASP3* (CASP activation)

CASP7* + CASP6 → CASP6* + CASP7* (CASP activation)

CASP3* + DFF → DFF45-CASP3* + DFF40 (DNA decomposition activation)

CASP7* + DFF → DFF45-CASP7* + DFF40 (DNA decomposition activation)

CASP3* + IAP → IAP-CASP3 (Inhibition)

CASP6* + IAP → IAP-CASP6 (Inhibition)

CASP7* + IAP → IAP-CASP7 (Inhibition)

Our simulation (Apte et al., 2010) has shown neither FLIP, nor XIAP could save the T-cells when acting alone. However, as shown in Fig. 6, when used together these inhibitors act synergistically, and could suppress the apoptosis almost entirely. A similar synergy trend shown to suppress apoptosis in type II colorectal cancer cells (Wilson et al., 2009) may be regarded as an indirect validation of our model.

An alternative, common strategy in fighting cancer is to use apoptosis to directly attack cancer cells. One of the way toward such a goal is to maximize the concentration of the "DNA killer" DFF40 by suppressing the apoptosis inhibitors FLIP and IAP. We simulated such a strategy by varying the transitional probability of the inhibitor suppressors siRNA and SMAC, respectively (Apte et al., 2010). Fig. 7 demonstrates that silencing FLIP, which is stronger inhibitor than IAP, does not suffice since the achieved active concentration of DFF40 does not exceed 60% of the theoretical maximum of 500 cells. The synergistic suppression of FLIP and IAP by siRNA and SMAC, respectively, raises this percentage to 90% and enables a full-scale apoptosis to kill the cancer cells.

We proceeded further from a more complete model of apoptosis by integrating the exogenous pathway of FASL-induced apoptosis with the endogenous pathway of mitochondria-activated apoptosis (Fig. 8). Cells undergoing apoptosis by these two mechanisms are called type I and type II, respectively (Chang et al., 2002; Wilson and al., 2009). The FASL-induced mechanism takes place at high levels of caspase-8, while low levels of this kinase result in expression of the protein BID, which activates the mitochondrial mechanism. The mitochondria releases cytochrome C into the cytoplasm, which in turn activates caspase-9. The cascade is closed with caspase-9 activating caspase-3. In addition, a feedback loop from caspase 3 to caspase 9 to IAP has been hypothesized to deactivate IAP (Creagh & Seamus, 2001; Zhou et al., 2005; Okazaki et al., 2009).

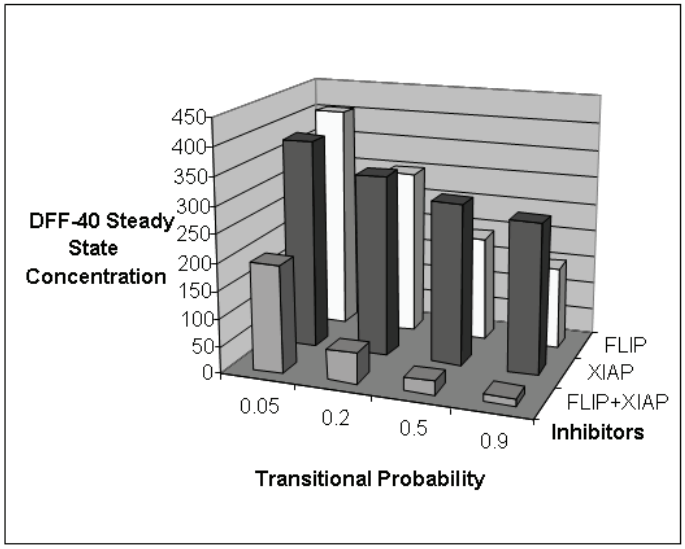


Fig. 6. The concentration of the DNA "killer" DFF-40 reduces with the increase of the inhibitor activity. Acting individually, FLIP and XIAP inhibitors cannot prevent the killing of the immune system T-cells by cancer cells and HIV infection. However, the CA simulation predicts a synergistic effect of the joint use of inhibitors that could save the T-cells, and restore the immune system potency.
(Cellular Automata (CA) Modeling of Biomolecular Networks Dynamics, D. Bonchev, S. Thomas, A. Apte, L. B. Kier, SAR & QSAR in Environmental Research, 2010, reprinted by permission of Taylor & Francis Ltd) http://www.informaworld.com).

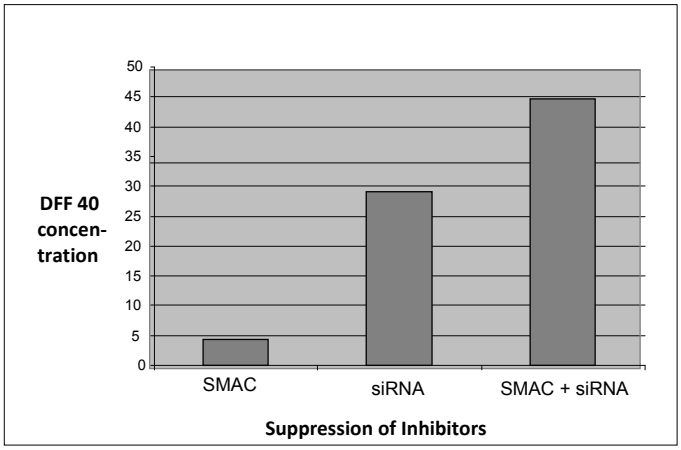


Fig. 7. Suppressing the FLIP and IAP inhibitors by siRNA and SMAC, respectively. The DFF40 steady-state concentration after 25000 cellular automata iterations predicts that a maximal FASL-induced apoptosis is achievable only via joint synergistic suppression of FLIP and IAP inhibitors. (Courtesy of "Chemistry and Biodiversity" journal (Apte et al., 2010-12-31)).

Our simulation showed that adding the feedback loop CASP3 → CASP9 → IAP to the mitochondria-mediated apoptosis pathway does not affect strongly the concentration of DFF40. However, the enhanced suppression of the IAP inhibitor and the additional activation of CASP9 accelerate considerably the process. We found that under these conditions FASL mechanism is 32 % faster than mitochondrial feed-forward mechanism, and 12 % faster than the mitochondrial feed-forward with a feed-back. (The number of iterations needed for the three mechanisms was 5012±12 vs. 5596±11 vs. 7368±13, respectively). The interconnectivity of the two apoptosis cascades thus offers a second, redundant mechanism for type-I cell apoptosis in case of failures in the FASL apoptosis pathway, such as no DISC formation or mutated membrane bound FAS, etc. Reducing the CASP8 concentration in such cases switches apoptosis to the mitochondrial pathway with feedback, which is only 12% slower. Complete details of the CA modeling and its parameters are given in (Apte et al., 2010). The data presented in this section demonstrate the great potential of cellular automata technique for biomedical applications.

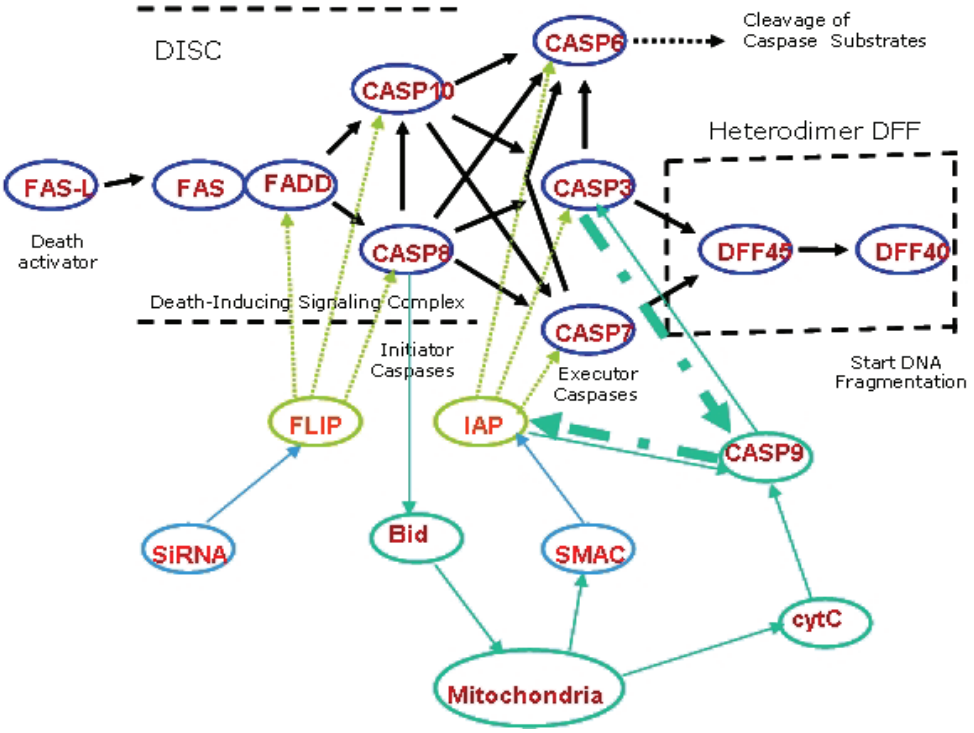


Fig. 8. Integrated scheme of the exogenous FASL-induced apoptosis and the endogenous mitochondrial apoptosis (Apte et al., 2010). At low expression level of CASP8 the mitochondria releases cytochrome C in the cytoplasm, activating thus CASP9, which in turn activates CASP3. A feedback from CASP3 to CASP9 increases the concentration of CASP9 active form, which accelerates apoptosis by suppressing the inhibition from IAP via SMAC protein. A similar suppression of the FAS apoptotic circuit can be achieved by using siRNA. (Courtesy of "Chemistry and Biodiversity" journal (Apte et al., 2010-12-31)).

5. CA modeling of network motifs performance

The cellular automata modeling discussed in the previous sections was directed toward identifying dynamic patterns in networks and pathways, which to provide means for control of their dynamics. A different approach was also pioneered in our Center for the Study of Biological Complexity (Bonchev et al., 2006, Apte et al., 2008, Taylor et al., 2010). It was aimed to search for answers to a fundamental problem: "*How structure affects the dynamics of processes in networks*?". While this question in the general case of networks of arbitrary size is too complex to be answered in a simple manner, a guiding idea was to look for exact answers for the dynamics of network motifs, the smallest structural units of networks (Milo et al., 2002; Alon, 2006, 2007). Such an approach avoids the computational complexity of large systems and, in addition, the CA derived patterns can be verified by ODE simulations (solutions of differential equations).

More specifically, in the search for the best performing structure we compared the same size motifs of different topology, and assessed their *performance* by the overall rate of the processes of conversion of the substrate(s) in the source node(s) into the product(s) of the motif target (or in terms of graph theory of the *subgraph sink*) node(s). In order to extract the topological factor chemical kinetics parameters (concentrations and rate constants) and probabilistic rules (except the transitional probability one), were kept constant. This "freezes" the process stochasticity, making our simulations *de facto* non-stochastic. Different classes of motif topology were defined according to the number of source (S) and target (T) nodes, with major attention being focused on the S1T1 class having a single source and a single target node. Our approach could be of particular interest for *signaling pathways* in biological systems, the overall rate of performance in which measures the effectiveness of converting the incoming chemical signal into an outgoing one. While using the language of biochemical reactions in describing motifs' links, and building on the specific CA approach to network discussed in the previous sections (Kier et al., 2005), the method can be readily applied to networks with different type of node-node relations, including ecological and social networks.

We performed a detailed analysis of the dynamics of different feed-forward (FF) motifs, which have been of considerable interest in biological systems. We extended the concept of FF motif used in the literature, namely "a subgraph that contains a feed-forward link connecting the source and the target nodes", to more general cases the added link in which shortens the distance between the source node and the target one, but not necessarily connects them directly. The *temporal* dynamics of feed-forward motifs in gene regulatory networks has been studied in detail (Mangan & Alon, 2003; Kashtan et al., 2004; Kashtan & Alon, 2005; Alon, 2006, 2007). It was shown that gene evolution depends on the topology of gene regulatory network (Cordero & Hogeweg, 2006). The relation between structural modules and dynamics of cellular networks, has been considered as a basis for cell reprogramming and engineering (Yuan & Hui, 2006). (Chechik et al., 2008) introduced activity and timing motifs, which capture patterns in the dynamic use of a network and reveal principles of transcriptional control of metabolic networks (Naemi, 2008). More generally, relating topology to function lead to a better understanding of dynamic properties of network motifs, e.g., their contribution to network stability (Prill et al., 2005).

Our approach is based on CA *spatial* models of the dynamics of generalized feed-forward motifs, which provide information on the concentrations of all substrates and products at the (quasi) steady-state reached after a considerable amount of CA iterations. The overall

reaction rate was assessed in parallel by CA and ODE simulations, as the number of iterations (respectively time in s) needed for 90% conversion of the input signal into the output one. The simulation was performed on a 2D-square lattice, 100 by 100 cells, embedded on the surface of a torus, with a lattice density of 3.6%. Each simulation was run 100 times, which produced a statistics with a sufficiently low standard deviation. Each motif's arc *i* was assumed to correspond to an enzymatic reaction: $S(i) + E(i) \rightarrow SE(i) \rightarrow PE(i) \rightarrow P(i) + E(i)$, and the probabilistic rules used were the ones for enzymatic reactions (See Section 3). The performance of all directed 4-node feed-forward motifs having a single source and a single target node (S1T1 class) was evaluated and compared to that of the directed linear motif of the same size.

The parallel ODE simulation was carried out in several approximations (Apte et al., 2008). The simplest way to construct an ODE model is to treat each feed-forward link A→B without regard to the underlying biochemical processes (e.g., neglecting the formation of substrate-enzyme and enzyme-product complex and the subsequent dissociation of the latter). In doing so, we neglect any nonlinear interactions of various species. The advantage of this linear ODE approach is that with the assumption for constant initial concentrations and rate constants, the linear systems of ODEs can be solved explicitly. Taking into account the formation of the substrate-enzyme complex *SE(i)*, and assuming that the substrate-enzyme complex *SE(i)* converts with a certain transitional probability into the product *P(i)* and a release of the enzyme *E(i)*, produced a nonlinear model (NDE), which can be solved only numerically. A second, more detailed nonlinear model (NDE´) has taken into account the reversibility of the process of formation of the intermediate SE(i) complex, which is a basic assumption in the theory of enzymatic reactions. The results summarized in Fig. 9 show very good agreement between CA and ODE models.

The ordering of the ten directed 4-node motifs in Fig. 9 was found to follow several topological transformation patterns (Apte et al., 2008). The acceleration of the S → T conversion might be predicted in part by conjecturing that every graph transformation that reduces the distance or, alternatively reduces the average path length, between the source and target vertices *S* and *T*, accelerates the process. Counting the distance between two neighboring vertices as a unit, one extracts from Fig. 9 a series of topological patterns that improve the motif dynamic performance.

Topodynamic Pattern 1: The shorter the graph distance d(*S*→*T*) between the source node and the target node in a feed-forward motif, the higher the overall motif dynamic performance:

$$\mathbf{A}(d = 3) < \mathbf{B}, \mathbf{C}\ (d = 2) < \mathbf{D}, \mathbf{E}, \mathbf{F}, \mathbf{G}, \mathbf{H}, \mathbf{I}\ (d = 1) \quad (2)$$

Notably, the two bi-parallel motifs **F** and **J** do not obey this pattern.

When considering the average path length one arrives at a more distinctive pattern. It singles out motif **I** to perform with the highest rate, due to the lowest average path length between nodes S and T (L = (1+2+2)/3 = 5/3):

Topodynamic Pattern 2: The shorter the average path length *L*(*S*→*T*) between the source node and the target node in a feed-forward motif, the higher the overall motif dynamic performance:

$$\mathbf{A}\ (L = 3) < \mathbf{B}, \mathbf{C}\ (L = 2.5) < \mathbf{D}, \mathbf{E}, \mathbf{G}, \mathbf{H}\ (L = 2) < \mathbf{I}\ (L = 1.67) \quad (3)$$

Topodynamic Pattern 3: Any ring closure of a linear chain of steps converting a source substrate *S* into a target product *T* accelerates the transformation. Acceleration of the process is the strongest when the feed-forward link directly connects the substrate to the target and is the smallest when the link connects the substrate to an intermediate product:

$$\mathbf{A < B < C < D} \tag{4}$$

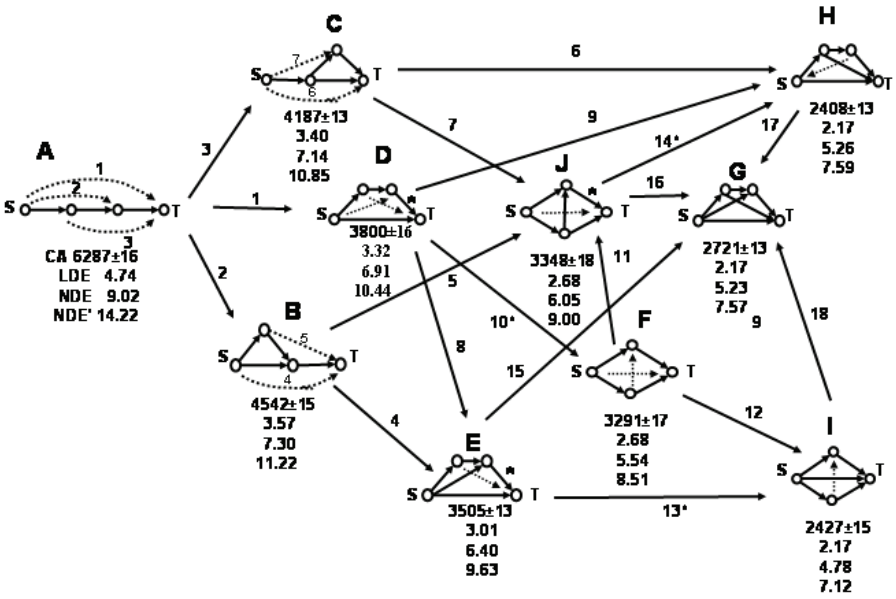
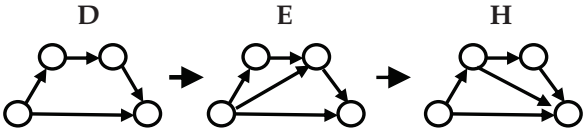


Fig. 9. Performance of 4-node network motifs, evaluated by the rate of converting the source node *S* substrate into the target node T product, as measured by the number of CA iterations, and by the time in seconds determined from a linear and two nonlinear differential equations models. The motifs from **A** through **J** correspond to ID numbers 536, 2118, 2076, 652, 2126, 2182, 2254, 2204, 2190 and 2140, respectively (Milo et al., 2002). The broken lines indicate the manner in which another directed link can be added in a subsequent topological transformation. The asterisks in motifs **D, E,** and **J**, stand for the edge, which changes its direction in a subsequent transformation. (Courtesy of Journal of Biological Engineering (Apte et al., 2008)).



	D	E	H
Iterations	3800±16	3505±13	2408±13
LDE	3.32	3.01	2.17
NDE	6.91	6.40	5.26
NDE´	10.44	9.63	7.59

Fig. 10. Adding a second feed-forward edge accelerates the motif performance, particularly when the edge is incident to the target node. (Courtesy of Journal of Biological Engineering (Apte et al., 2008)).

Topodynamic Pattern 4: Adding a second feed-forward edge (*double feed-forward motif*), between a pair of nodes in the longer path of the FF loop, accelerates the conversion of the source substrate into the target product:

$$D < E < H \tag{5}$$

The pattern is illustrated in Fig. 10. Adding a third feed-forward edge does not always have an accelerating effect, as seen from the motifs **H** → **G** transformation.

Topodynamic Pattern 5: Reversing the direction of one or more links in a feed-forward motif to turn it into a bi-parallel and tri-parallel one increases the motif performance:

$$\textbf{Feed-Forward < Bi-Parallel < Tri-Parallel} \tag{6}$$

Three such conversions:

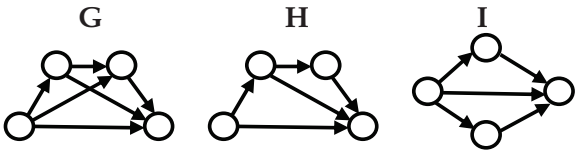
$$\textbf{D < F, E < I, J < H} \tag{7}$$

are shown in Fig. 9, where they are denoted by asterisks.

Topodynamic Pattern 6 (Isodynamicity): Some feed-forward motifs with different topology are characterized by the same overall $S \rightarrow T$ conversion rate by the CA and linear ODE models:

$$\textbf{CA: H (2408 ± 13) = I (2427 ± 15)} \tag{8a}$$

$$\textbf{ODE: G = H = I = 2.169053700 s} \tag{8b}$$



2721±13	2408±13	2427±15
2.17	2.17	2.17
5.23	5.26	4.78
7.57	7.59	7.12

Fig. 11. Motifs **G**, **H**, and **I** are isodynamic according to linear ODE model, whereas CA models confirms the isodynamicity of **H** and **I**. The two nonlinear models show very close isodynamicity of **G** and **H**, while the more detailed NDE´ model singles out motif **I** as the best performing one, as also predicted by purely topological arguments in Topodynamic Pattern 2 (see above).

Equality (8a) is valid within the standard deviation ranges of **H** and **I** (2395-2421 vs. 2412-2442). The linear ODE's times also characterize motifs **F** and **J** as isodynamic, whereas their CA estimates diverge slightly (3274-3308 vs. 3330-3366, respectively). The two nonlinear NDE times of motifs **G** and **H** are very close, while the most complex NDE´ model classifies motif **I** as best performing:

$$\textbf{NDE: G (5.23) ≈ H (5.26) > I (4.78)} \tag{9a}$$

$$\textbf{NDE´: G (7.57) ≈ H (7.59) > I (7.12)} \tag{9b}$$

The concept of motifs isodynamicity was investigated in more details by linear ODE models. Three theorems were proved (Taylor et al., 2010) for classes of motifs sharing this property. The first such class is motifs containing target vertex with maximal in-degree (Fig. 12a):

Theorem 1. Consider the family of feed-forward motifs on n vertices with a single target vertex. Then all motifs for which the in-degree of the target vertex is (n−1) are isodynamic.

This theorem is easily extended to motifs having many target nodes (Fig. 12b):

Theorem 2. Suppose $1 < k < n$ and consider the family of feed-forward motifs on n vertices with precisely k target vertices. Then all motifs for which the in-degrees of the target vertices are (n−k) are isodynamic.

Theorem 1 expands the isodynamicity pattern so as to incorporate the class *S(n-1)T1*, while Theorem 2 expands that pattern further to the class of motifs *S(k)T(n-k).* The third theorem defines isodynamicity in a class of bi-parallel motifs. This class is also of S1T1 type but the single source and single target nodes are connected by two parallel chains of links. Adding in a specific manner links between the two parallel chains does not change the overall motif performance (Fig. 12c).

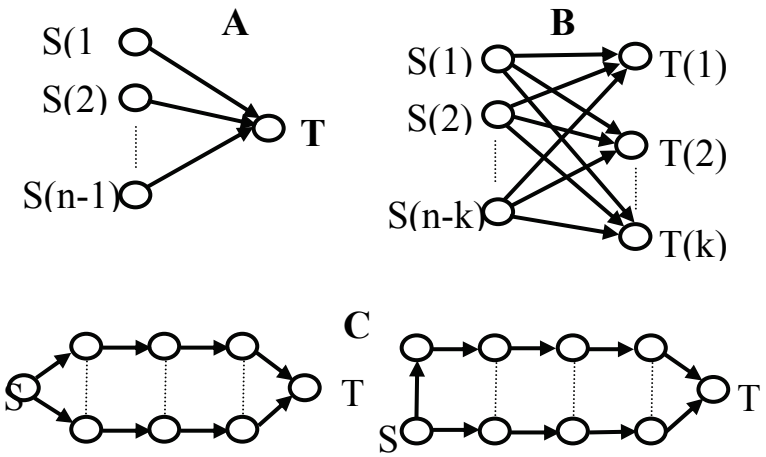


Fig. 12. A, B, C. Illustration of Theorems 1, 2 and 3, respectively.

Theorem 3. Consider the bi-parallel motif on m vertices, with the alternating vertex labeling. Suppose we construct a new motif by adding directed edges between vertices k and (k+1) (regardless of orientation) if k has the same parity as m and $1 < k < (m-1)$. Then this new motif is isodynamic with the bi-parallel motif.

Theorems 1-3 thus identified two large classes of isodynamic feed-forward motifs: such the target vertices of which have maximal in-degree, and bi-parallel motifs with a variable number of redundant edges not changing the performance rate. An idealized case was used, all reactions in which proceed at the same rate, and the formation of intermediate complexes is not taken into account. Nevertheless, numerical simulations with more realistic nonlinear ODE models have shown time estimates close to those produced by the linear models and the CA ones.

6. Conclusion

This chapter summarizes the pioneering work on cellular automata modeling of network dynamics done in our Laboratory. Although obtained at molecular biology level, the findings of our study are easily applicable to complex networks of arbitrary nature. Our approach to such level of complexity is to study in detail small size subnetworks (motifs), reducing strongly computational time, while shedding light on the dynamic patterns of the system as a whole. This approach does not provide exact answers, but rather identifies patterns of behavior. It offers answers to questions like what one has to expect when affecting a network node or node-node interaction (link), providing thus means for network control. The latter could facilitate the search for novel pharmacological targets, as well as for individualized patient treatments. As shown in our work, intimate details of the mechanism of action of diseases can be revealed, such as cooperative action of enzymes, synergetic action or suppression of inhibitors, etc. All this information provides a basis for developing strategies for fighting such diseases like cancer and HIV.

An essential part of such studies is the extraction of useful topological-dynamic (topo-dynamic) patterns describing specific effects of topological structures on network dynamics at constant other conditions. The great advantage of using topology to study network dynamics is in the generality of the patterns found, which do not depend on process specificity or network size. The dynamics of the feed-forward motifs investigated revealed important aspects of networks containing such loops. Any feed-forward link added to a linear cascade of chemical/ biochemical reactions accelerates the process, and the acceleration is further enhanced by adding a second feed-forward link. The acceleration of the overall process in FF motifs increases with the decrease in the distance, and in the average path length, between the input and output nodes. When the distance parameters are kept constant, cellular automata and ODE simulations produce a further finer distinction between the motifs dynamic performance. The concept of isodynamic network motifs revealed important aspects of similarity in dynamic behavior of subnetworks of different topology. The consequence for biological and other systems from this finding is that identical or closely similar rate of performance of processes converting a given input to a desired output can be produced by different network connectivity. It is important to understand whether there is a specific selective advantage to use a certain motif topology among a number of others of similar performance rate.

In the more general case of non-isodynamic network motifs one may expect that evolution might have been using the higher speed of producing a desirable target product from equivalent initial conditions, particularly in signaling pathways. The answer of this question is a subject of our extensive almost completed study, in which the abundance of motifs in metabolic networks, and their higher level of organization termed network of interacting pathways (NIP) (Mazurie et al., 2008, 2010), were analyzed in over 1000 species. Evidence for high statistical support was recovered for the over-representation of certain feed-forward and bi-parallel motifs (subgraphs) with 3 and 4 nodes. The motifs exhibiting considerable enrichment were those having faster performance dynamics and extra null-performance link. The preliminary results favored strongly one of the three fastest motifs found (motif ID # 2204, denoted as **G** in Fig. 9). The high abundance of this motif evidences that evolution conserves this effective topology of maximum cross-talk between the individual metabolic pathways. Motif 2204 exhibits the additional advantage to keep its overall performance almost unchanged even in case of losing one of its links (null-

performance link), in which case it converts to the tri-parallel motif **I** (ID # 2140), which is also one of the three fastest performing motifs. The lack of statistically significant abundance of such a high speed subgraph may be interpreted as evidence that at equal or close efficacy evolution conserves the structure that provides a higher stability. Having an extra edge which does not contribute to a higher conversion rate is a beneficial redundancy; if this edge is destroyed or incapacitated, the efficacy of performance of the biochemical reactions will remain practically the same. In the context of adaptive significance, these results indicated that the need of higher network resilience against attacks not only compensates the energy price for the extra link formation but also exceeded the potential benefit of a faster performance. Further studies extend this type of motif dynamics analysis on *Drosophila* microRNA-target interaction networks (Woodcock, 2010).

7. Acknowledgments

The contributions of Drs. L. B. Kier, A. Apte, C.-K. Cheng, J. W. Cain, D. Taylor, S. Fong, and L. E. Pace to the publications summarized in this chapter are highly acknowledged.

8. References

Adrain, C., Creagh, E. M. & Seamus, J. (2001). Apoptosis-associated release of Smac/DIABLO from mitochondria requires active caspases and is blocked by Bcl-2. *EMBO J*. 20, 6627-6636.

Albert, R., Jeong, H. & Barabási, A.-L. (2000). Error and attack tolerance of complex networks, *Nature*, 406 (July 2000), 378-382.

Albert, R. and Barabási, A.-L. (2000). Topology of complex networks: Local events and universality, *Phys. Rev. Lett.* 85, No. 24, (Dec 2000) 5234-5237.

Aldana, M. (2003). Boolean dynamics of networks with scale-free topology. *Physica D*, 185, 45–66.

Alon, U. (2006). *An introduction to Systems Biology: Design Principles of Biological Circuits*, Chapman & Hall/CRC, ISBN: 1-58488-642-0, Boca Raton, FL.

Alon, U. (2007) Network motifs: theory and experimental approaches. *Nature Rev. Genet.* 8, (June 2007), 450-461.

Apte, A., Cain, J. W., Bonchev, D. & Fong, S. S. (2008). Topological effects on the dynamics of feed-forward motifs, *J. Biol. Eng*. 2, 2-13.

Apte, A., Bonchev, D. & Fong, S. S. (2010) Cellular automata modeling of FAS-initiated apoptosis, *Chem. Biodiversity* 7, 1163-1172.

Barabási, A.-L. & Albert, R. (1999). Emergence of scaling in random networks. *Science* 286, 509–512.

Barabási, A.-L. (2004). *Linked: How Everything is Connected to Everything Else*, Perseus, ISBN 0-452-28439-2, Cambridge, MA.

Barabási, A.-L. & Oltvaj, L. Z. (2004). Network biology: Understanding the cell's functional organization. *Nature Genet*. 5, 102-114.

Barrat, A., Barthélemy, M., Pastor-Satorras, R. & Vespignani, A. (2004). The architecture of complex weighted networks. Proc. Nat. Acad. Sci. 101 No. 11, (Mar 2004) 3747–3752.

Bhalla, U. S. & Iyengar, R. (1999). Emergent properties of networks of biological signaling pathways. *Science*, 283, (Jan 1999), 381-387.

Bollobás, B. (2001). *Random Graphs*. Cambridge University Press, IBSN: 0-521-80920-7, Cambridge, U. K.

Bonchev, D., Kier, L. B. & Cheng, C.-K., (2006). Cellular automata (ca) as a basic method for studying network dynamics. *Lecture Series on Computer and Computational Sciences* 6, 581-591.

Bonchev, D., Thomas, S., Apte, A. & Kier, L. B. (2010). Cellular automata (CA) modeling of biomolecular networks dynamics. *SAR & QSAR Envir. Res*. 21, 77–102.

Borgatti, S. P., Mehra, A., Brass, D. & Labianca, G. (2009). Network analysis in the social sciences. *Science*, 323, No. 5916 (Feb 2009) 892- 895.

Brajušković, G. R. (2005). Apoptosis in malignant diseases. *Arch. Oncol.* 13, No. 1, (Sept 2005), 19-22.

Butcher, E. C., Berg, E. L. & Kunkel, E. J. (2004). Systems biology in drug discovery. *Nat. Biotechnol. 22, (Oct 2004)*, 1253 – 1259.

Chang, D. W., Xing, Z., Pan, Y., Algeciras-Schimnich, A., Barnhart, B. C., Yaish-Ohad, S., Peter, M. E. & Yang, X. (2002). c-FLIP$_L$ is a dual function regulator for caspase-8 activation and CD95-mediated apoptosis. *EMBO* EMBO J., 21, 3704-3714.

Chechik, G., Oh, E., Rando, O., Weissman, J., Regev, A. & Koller, D. (2008). Activity motifs reveal principles of timing in transcriptional control of the yeast metabolic network. *Nature Biotechnol. 26*, (Oct 2008) 1251 - 1259.

Cheung, H. H., Mahoney, D.J., LaCasse, E. C. & Korneluk, R. G. (2009). Death of c*ancer* cells by smac mimetic compound. Cancer Res., 69, 7729-7738.

Choi, S. (2007). *Introduction to Systems Biology*. Humana Press, ISBN: 978-1-58829-706-8, Totowa, NJ.

Cordero, O. X. & Hogeweg, P. (2006). Feed-forward loop circuits as a side effect of genome evolution. *Mol. Biol. Evol.* 23, No. 10, (Jul 2006) 1931-1936.

Csikasz-Nagy, A., Battogtokh, D., Chen, K. C., Novak, B., & Tyson, J. J. (2006). Analysis of a generic model of eukaryotic cell-cycle regulation. *Biophys. J.* 90, No. 12, (Mar 2006) 4361-4379.

The Cytoscape software, http://cytoscape.org/, UCSD, San Diego, CA.

Dorogovtsev, S. N., Mendes, J. & Samukhin, A. (2000). Structure of growing networks: Exact solution of the Barabasi-Albert model. *Phys. Rev. Lett*. 85, 4633-4636.

Dorogovtsev, S. N. & Mendes, J.F.F. (2003). *Evolution of Networks: From biological networks to the Internet and WWW*, Oxford University Press, ISBN 0-19-851590-1, Oxford, U. K.

Ermentrout, G. B. & Edelstein-Keshet, L. (1993). Cellular automata approaches to biological modeling, *J. Theor. Biol*. 160, No. 1, (Jan 1993) 97-133.

Friedkin, N. E. (1984). Structural cohesion and equivalence explanations of social homogeneity. *Sociol. Meth. Res*., 12, 35–61.

Fadeel, B. & Orrenius, S. (2005). Apoptosis: a basic biological phenomenon with wide-ranging implications in human disease. *J. Intern. Med.*, 258, No. 6, (Oct 2005), 479–517.

Ferguson, T. A. & Griffith, T. S. (2006). A vision of cell death: Fas ligand and immune privilege 10 years later. *Immunol. Res.* 213, No. 1, (Oct 2006) 228-238.

Friedkin, N. E., & Johnsen, E. C. (1990). Social influence and opinions. *J. Math. Sociol*. 15, 193–206

Ideker, T. (2004). Systems Biology 101 – What you need to know. *Nat. Biotechnol*. 22, 473-475.

Fulda, S. & Debatin, K. M. (2004). Apoptosis signaling in tumor therapy. *Ann. N. Y. Acad. Sci.*, 1028, No.1, (Jan 2006),150-156.

Gardner, M. (1970). Mathematical Games: The fantastic combinations of John Conway's new solitaire game "life". *Sci. Amer.* 223, (Oct 1970) 120-123.

Gene Ontology, http://www.geneontology.org/GO.database. shtml).

Guimera, R. & Amaral, L. A. N. (2005). Functional cartography of complex metabolic networks. *Nature*, 433, 895-900.

Hanahan, D. & Weinberg, R. A. (2000). The hallmarks of cancer. *Cell*, 100, No. 1, (Jan 2000) 57-70.

Hintze, A. & Adami, C. (2008). Evolution of complex modular biological networks. *PLoS. Comput. Biol*. 4, No. 2, e23.

Hoffman, P. (1998). *The Man Who Loved Numbers: The Story of Paul Erdos and the Search for Mathematical Truth.* Hyperion, ISBN. 0-786-86362-5, New York, NY.

Hopkins, A. L. (2008). Network pharmacology: the next paradigm in drug discovery, *Nat. Chem. Biol. 4, (Oct 2008)* 682-690.

Huang, C.-Y. F. & Ferell, J. E. (1996). Ultrasensitivity in the mitogen-activated protein kinase cascade, *Proc. Natl. Acad. Sci. USA*, 93, (Sep 1996) 10078-10083.

Human Proteome Organization (HUPO) www.hupo.org.

Igney, F. H. & Krammer, P. H. (2005). Tumor counterattack: fact or fiction? Cancer Immunol. Immunother. 54, No. 11, 1127-1136.

Ingenuity Pathways Analysis (Ingenuity® Systems, www.ingenuity.com).

Irmler, M., Thome, M., Hahne, M., Schneider, P., Hofmann, K., Steiner, V., Bodmer, J. L., Schroter, M., Burns, K., Mattmann, C., Rimoldi, D., French L. E. & Tschopp, J. (1997). Inhibition of death receptor signals by cellular FLIP. *Nature*, 388, 190-195.

Jin, Z. & El-Deiry, W. S. (2005). Overview of cell death signaling pathways. *Cancer Biol. Ther.*, 4, No. 2, (Feb 2005) 139-163.

Kashtan, N., Itzkovitz, S., Milo, R. & Alon, U. (2004). Topological generalizations of network motifs. *Phys. Rev. E*, Vol. 70, No. 3, 031909(12).

Kashtan, N. & Alon, U. (2005) Spontaneous evolution of modularity and network motifs. *PNAS*, 102, No. 39, (Sep 2005) 13773-13778.

Kauffman, S. (1969). Metabolic stability and epigenesis in randomly constructed genetic nets. *J. Theor. Biol*., 22, 437–467.

Kauffman, S. (1993). *The Origins of Order: Self-Organization and Selection in Evolution*. Oxford University Press, ISBN: 0-19-505811-9, New York, NY.

Kauffman, S., Peterson, C., Samuelsson, B. & Troein, C. (2003). Random Boolean network models and the yeast transcriptional network. *PNAS*, 100, No. 25, (Dec. 2003) 14796--14799.

KEGG, http://www. genome. jp/kegg/kegg1.html.

Kholodenko, B. N. (2000). Negative feedback and ultrasensitivity can bring about oscillations in the mitogen-activating kinase cascades. *Eur. J. Biochem.*, 267, (Feb 2000) 1583-1588.

Kier, L. B., C.-K. Cheng, B. Testa & P.-A. Carrupt (1996). A cellular automata model of enzyme kinetics. *J. Molec Graphics*, 14, (Aug 1996) 227-231.

Kier, L. B., Seybold, P. G. & Cheng, C.- K. (2005a). *Cellular Automata Modeling of Chemical Systems*. Springer, ISBN: 1-4020-3657-4, Amsterdam.

Kier, L. B. & Witten, T. M. (2005b). Cellular automata models of complex biochemical networks, In: *Complexity in Chemistry, Biology and Ecology,* Bonchev, D. & Rouvray, D. H. (Ed.), 237-301, Springer, ISBN: 0-387-23264-8, New York, NY.

Kier, L. B., Bonchev, D. & Buck G. A. (2005c). Modeling biochemical networks: A cellular automata approach. *Chem. Biodiversity* 2, No. 2, (Feb 2005) 233-243.

Kitano, H. (2001). *Foundations of Systems Biology*, MIT Press, ISBN-13:978-0-262-11266-6, Cambridge, MA.

Kitano, H. (2002). Computational systems biology. *Nature* 420, 206–210.

Knabe, J. F., Nehaniv, C. L. & Schilstra, M. J. (2008) Do Motifs Reflect Evolved Function? – No. Convergent Evolution of Genetic Regulatory Network Subgraph Topologies. *BioSystems* 94, No. 1-2, 68-74.

Konagurthu, A. S. & Lesk, A. M. (2008). On the origin of distribution patterns of motifs in biological networks. *BMC Syst. Biol.* 2, (Aug 2008) 73(8).

Lavrik, I. N., Eils, R., Fricker, N., Pforr, C. & Krammer, P. H. (2009). Understanding *apoptosis* by systems biology approaches. *Molec. BioSyst*., 5, No. 8, (Aug 2009) 1105-1111.

Longabaugh, W.& Bolouri, H. (2006). Understanding the dynamic behavior of genetic regulatory networks by functional decomposition. *Curr. Genomics*, 7, No. 6, (Nov 2006) 333–341.

Lowe, S. W., Cepero, E. Evan, & G. (2004). Intrinsic tumor suppression. *Nature*, 432, (Nov 2004), 307-315.

Managbanag, J. R., Witten, T. M., Bonchev, D., Fox, L. A., Tsuchiya, M., Kennedy, B. K. & Matt Kaeberlein (2008). Shortest-path network analysis is a useful approach toward identifying genetic determinants of longevity. *PLoS ONE* 3, No. 11, (Nov 2008) e3802.

Mangan, S. & Alon, U. (2003). Structure and function of the feed-forward loop network motif. *Proc. Natl. Acad. Sci. USA*, 100, No. 21, (oct 2003) 11980–11985.

Marek, L., Burek, C. J., Stroh, C., Benedyk, K., Hug, H. & Mackiewicz, A. (2003). Anticancer drugs of tomorrow: apoptotic pathways as targets for drug design. *Drug Discov. Today*, 8, No. 2, 67-77.

Mazurie, A., Bonchev, D., Buck, G. A. & Schwikowski, B. (2008). Phylogenetic Distances Are Encoded in Networks of Interacting Pathways. *Bioinformatics*, 24, No. 22, (Sep 2008) 2579-2585.

Mazurie, A., Bonchev, D., Schwikowski, B. & Buck, G. A. (2010). Evolution of metabolic networks organization, *BMC Systems Biol*ogy 4, No. 2, 59-68.

McCullagh, E., Seshan, A., El-Samad, H. & Madhani, H. D. (2010). Coordinate control of gene expression noise and interchromosomal interactions in a MAP kinase pathway. *Nat. Cell Biol*. 12, No. 10, 1-11.

Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D. & Alon U. (2002). Network motifs: simple building blocks of complex networks. *Science*, 298, No. 5594, (Oct 2002) 824-827.

Nadeau, J. H. & Subramaniam, S. (2010). Wiley Interdisciplinary Reviews: Systems Biology and Medicine, Wiley Periodicals, Inc., Online ISSN: 1939-005X.

Neame, E. (2008). Gene networks: Network analysis gets dynamic. *Nature Rev. Genetics* 9, (Dec 2008) 897.

Newman, M.E.J. (2003). The structure and function of complex networks. *SIAM Review* 45, 167–256.

Newman, M. E. J. & Girvan, M. (2004). Finding and evaluating community structure in networks, *Phys. Rev. E* 69, 026113.

Newman, M., Barabasi, A.-L. & Watts, D. J. (2006). *The Structure and Dynamics of Networks*. Princeton University Press, ISBN: 978-0-691-11357-9, Princeton, NJ.

Neuman, M. E. J. (2006). Modularity and community structure in networks. *PNAS*, 103, No. 23, (Jun 2006) 8577-8582.

Okazaki, N., Asano, R., Kinoshita, T. & Chuman, H. (2008). Simple computational models of type I/type II cells in Fas signaling-induced apoptosis. *J. Theor. Biol.* 50, No. 4, 621-633.

Opsahl, T. & Panzarasa P. (2009). Clustering in weighted networks. Soc. Networks, 31, No. 2, 155–163.

Palsson, B. (2006). Systems Biology: Properties of Reconstructed Networks. Cambridge University Press, ISBN: 0521859034, Cambridge, MA.

Pathway Studio (Ariadnegenomics, Inc., www.ariadnegenomics.com).

Peak, D., West, J. D., Messinger, S. & Mott, K. A. (2004). Evidence for complex, collective dynamics and distributed emergent computation in plants. *Proc. Nat. Acad. Sci.*, 101, No. 4, (Jan 2004) 918–922.

PINA, http://csbi.ltdk.helsinki.fi/pina/.

Prill, R. J., Iglesias, P. A. & Levchenko, A. (2005). Dynamic properties of network motifs contribute to biological network organization. *PLoS Biol.*, 3, No. 11, (Nov 2005) e343.

Yuan, Q. & Hui, G. (2006). Modularity and dynamics of cellular networks. *PLoS Comput. Biol.* 2006, 2, No. 12, (Dec 2006) 1502-1510.

Reed, J. C. (2006). Drug Insight: cancer therapy strategies based on restoration of endogenous cell death mechanisms. *Nat. Clin. Pract. Oncol.*, 3, No. 7, (Apr 2006), 388-398.

Reichardt, J. & Bornholdt, S. (2006). Statistical mechanics of community detection. Phys. Rev. E 74, (Jul 2006) 016110.

Rives, A. W. & Galitski, T. (2003). Modular organization of cellular networks. *PNAS*, 100, No. 3, (Feb 2003) 1128–1133.

Rucker, R. (2005). *The Lifebox, the Seashell, and the Soul*, Thunder's Mouth Press, ISBN 1-560-25722-9, New York, NY.

G. S. & Riedl, S. J. (2009). Structure of the Fas/FADD complex: A conditional death domain complex mediating signaling by receptor clustering. *Cell Cycle*, 8, No. 17, (Sep 2009), 2723-2727.

Scott, F. L., Stec, B., Pop, C., Dobaczewska, M., Klee, J. E. J., Monosov, E., Robinson, H., Salvesen, G. S., Schwarzenbacher, R. & Riedl, S. J. (2009). The Fas–FADD death domain complex structure unravels signaling by receptor clustering. *Nature,* 457, No. 7232, (Feb 2009) 1019-1022.

Scott, J. (1991). *Social network analysis. A Handbook.* Sage Publications, ISBN 0-7619-6338-3, London, U. K.

Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., Amin, N., Schwikowski, B. & Ideker, T. (2003). Cytoscape: A software environment for integrated models of biomolecular interaction networks. *Genome Res.* 13, (Oct 2008), 2498-2504. http:// cyto scape. org/ download.php.

Strogatz, S. H. (2001) Exploring complex networks. *Nature*, 410, (Mar 2001), 268–276.

Taylor, D. T., Cain, J. W., Bonchev, D., Apte, A., Fong, S. S. & Pace, L.E. (2010), Toward a classification of isodynamic feed-forward motifs, *J. Biol. Dynamics*, 4, No. 2, (Mar 2010) 196-211.

Thomas, S. & Bonchev, D. (2010). A survey of current software for network analysis in molecular biology, *Human Genomics,* 4, No. 5, (Jun 2010) 353-360.

Tyson, J. J., Chen, K. & Novak, B., (2004). Network dynamics and cell physiology, *Nature Rev. Molec. Cell Biol*., 2, No. 12, (Dec 2001) 908-916.

Vazquez, A. (2003). Growing network with local rules: preferential attachment, clustering hierarchy, and degree correlations. *Phys Rev E*, 67, (May 2003) 056104.

von Neumann, J. (1966). *Theory of Self-Reproducing Automata*, Burks, A.W. (Ed.), Univ. of IL Press, Urbana, IL.

Watts, D. J. & Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, 393, (Jun 1998) 440–442.

Watts, D. J. (1999). Networks, dynamics, and the small-world phenomenon". *Amer. J. Sociology*, 105, No. 2, (Sept 1999) 493–527.

Watts, D. J. (2003) *Small Worlds: The Dynamics of Networks between Order and Randomness*, Princeton University Press, ISBN 0-691-11704-7, Princeton, NJ.

Wajant, H. (2002). The Fas signaling pathway: more than a paradigm. Science, 296, No. 5573 (May 2002), 1635 - 1636.

Weimar, J. R. (2002). Cellular automata approaches to enzymatic reaction networks. Fifth International Conference on Cellular Automata for Research and Industry ACRI, Geneva, October 2002. *Lecture Notes in Computer Science* 2493, 294-303, Springer, Berlin.

Weitz, J. S., Benfey, P. N. & Wingreen, N. S. (2007). Evolution, interactions, and biological networks. *PLoS Biol*., 5, No. 1, (Jan 2007) e11.

Wernicke, S. & Rasche, F. (2006). FANMOD: a tool for fast network motif detection, B*ioinformatics*, 22, 52-53.

Wilson, T. R., McEwan, M., McLaughlin, K., Le Clorennec, C., Allen, W. L., Fennell, D. A., Johnston, P. G. & Longley, D. B. (2009). Combined inhibition of FLIP and XIAP induces Bax-independent apoptosis in type II colorectal cancer cells. *Oncogene*, 28, No. 28, (Jan 2009) 63-72.

Wolfram, S. (1986) *Theory and Applications of Cellular Automata,* Advanced Series on Complex Systems, World Scientific, Singapore, 1986.

Wolfram, S. (2002). *The New Kind of Science*, Wolfram Media, ISBN 1-57955-008-8, Champaign, IL.

Wong, S. L., Zhang, L. V., Tong, A. H. Y., Li, Z., Goldberg, D. S., King, O. D., Lesage, G., Vidal, M., Andrews, B., Bussey, H., Boone, C., Roth, F. P. & Kleckner, N. (2004). Combining biological networks to predict genetic interactions. *PNAS*, 101, No. 44, (Nov 2004) 15682-15687.

Woodcock M. R. (2010). Network analysis and comparative phylogenomics of microRNAs and their respective messenger RNA targets using twelve *Drosophila* species. Ph. D. Thesis (Nov 2010).

Zhigulin, V. P. (2004). Dynamical motifs: building blocks of complex network dynamics. *Phys. Rev. Lett.* 92, No. 23, (Jun 2004) 238701.

Zhou, L. L., Zhou, L. Y., Luo, K. Q. & Chan, D. C. (2005). Smac/DIABLO and cytochrome *c* are released from mitochondria through a similar mechanism during UV-induced apoptosis. *Apoptosis*, 10, 289-299.

14

Simulation of Qualitative Peculiarities of Capillary System Regulation with Cellular Automata Models

G. Knyshov, Ie. Nastenko, V. Maksymenko and O. Kravchuk
National M. Amosov Institute of Cardio-Vascular Surgery,
Dept. of Computer Sciences and Computational Physiology
National Technical University of Ukraine "KPI", Kyiv,
Ukraine Faculty of Biomedical Engineering
Ukraine

1. Introduction

Peripheral circulation can be divided in two components: continuous one, which includes continual fluctuation of muscular vessels diameters; and discrete one, which is caused by opening and closing of pre-capillary sphincters causes the activation-deactivation of capillaries. The state of each capillary sphincter depends on concentration of tissues vasodilator metabolites in the neighborhood of this capillary.

Application of Cellular Automata simulation approach allows to study the behavior of separate capillary system as critically self-organized system and to investigate the different types of interaction between muscular arterial vessels and capillary network.

Analysis of clinical data revealed significant difference in oxygen concentration in arterial and venous blood at normal conditions, at blood flow insufficiency conditions, and at hyperdynamic state of system circulation. This causes significant influence on fluctuation properties of capillary network and of tissues oxygen saturation. Depending on the state of system blood flow each capillary opens and closes at different thresholds of concentration of tissues vasodilator metabolites, closely correlated with tissues oxygen content.

Different types of capillary network behavior have been simulated for various heart rhythm disturbances, which cause irregularity of fluctuations of system blood flow.

Results of simulations were collated with clinical data and have a good correspondence with them.

Regulation of peripheral circulation of blood includes its metabolically caused redistribution, stabilization of the main hemodynamic characteristics at the level of microcirculatory units and mass transfer inside them. The final stabilization of blood pressure occurs at the level of arterioles, then in microcirculatory unit performs a counter-transport of oxygen and metabolites through the wall of the true capillary and arterio-venous shunting of blood (Chernukh&Alexandrov, 1984; Rushmer, 1986, e.a.).

The blood flow in arteries is studied in most detail (Achakri e.a. 1994, Cavalcanti & rsino, 1996; Little, 1989; Rushmer, 1986, e.a.). However, the integral properties of peripheral blood

circulatory system is not been adequately studied both at the empirical level, and theoretically, using mathematical models.
The aim of this study is to build a model of microcirculatory network in the form of a cellular automaton, based on information about the anatomy and principles of functioning of the system, to investigate its basic static and dynamic properties and to compare the results with the data from clinical investigations.
Despite the highly stable conditions of blood flow in single capillary, the number of active, working (opened) capillaries is variable and is determined by local metabolic activity of tissues. According to the literature (Achakri e.a. 1994; Chernukh, Alexandrov, 1984; Rushmer, 1986, e.a.), duration of active state of the capillary was 20-70s. That's why capillary network is usually described as a highly inertial, homeostatic, conservative system. However the stability of blood flow conditions in a single capillary can be erroneously identified with the constancy of systemic capillary blood flow in general, which does not correspond to the physiological reality. Bacause the quantity of active capillaries is essentially variable.
In accordance with the concepts of nonlinear science (synergetics) concerning to critically self-organized systems (Bak e.a., 1987,1996; Yusupov, Polonnikov, 1998), the capillary network can be considered as a large interactive system functioning at the "edge of chaos", or, following the terminology of (Risk management, 2000), in a "stable disequilibrium". Regulation of tissue blood flow must be as dynamic as possible to maintain the oxygenation of tissue on appropriate level, which can be inherent in systems with a critical self-organization.
From this point of view, a large variability of quantity of active capillaries can be explained from supposition that a large number of them being in near-critical, most sensitive to any influences, states.
Lack of empirical information can be, at least partially, overcome with modern methods of mathematical modeling using cellular automata simulations.

2. Clinical data analysis

2.1 Initial clinical data

716 obs. collected in intensive care unit (ICU) in 1-2 days after heart valve replacement and/or coronary-aortic bypass grafting (CABG).
The cardiac index (*CI*), systolic, diastolic, mean (*MAP*) arterial and central venous (*CVP*) pressures, systemic vascular resistance index (*SVRI*), central body temperature, indices of oxygen delivery (IDO_2) and consumption (IVO_2), oxygen content in arterial (Co_{2a}) and venous (Co_{2v}), their arterio-venous gradient (ΔCo_{2av}) pHa,v in arterial and venous blood and some other biochemical parameters were studied.
It should be mentioned that in order to provide the comparability of the indices of oxygen transport, they were normalized to the body surface area.
Cardiac index was calculated by the formula (Kaplan, 1979Ream & Fogdal, 1982, e.a.):

$$CI = CO \,/\, BSA, \tag{1}$$

where: *CO*, l/min – cardiac output, *BSA*, m^2 – patient body surface area.
Indices of systemic oxygen delivery (IDO_2) and consumption (IVo_2) normalized to body surface area (ml/(min • m^2)) and calculated by the formulas (Reeder, 1986; Samsel & Shumacker, 1981, e.a.):

$$IDO_2 = Co_{2a} \bullet CI; \tag{2}$$

$$IVO_2 = (Co_{2a} - Co_{2v}) \bullet CI, \tag{3}$$

Where: CI, l/(min • m²) - cardiac index; Co_{2a}, ml/dl - oxygen content in arterial blood; Co_{2v}, ml/dl - oxygen content in venous blood.
Oxygen content (concentration) in arterial and venous blood were determined by the equations (Naylor-Sheferd e.a., 1990):

$$Co_{2a} = SO_{2a} \bullet Hb \bullet 1{,}39\ (100 \bullet a) \ / \ Patm \bullet pO_{2a} \tag{4}$$

$$Co_{2v} = SO_{2v} \bullet Hb \bullet 1{,}39\ (100 \bullet a) \ / \ Patm \bullet pO_{2v}, \tag{5}$$

where Hb - hemoglobin; SO_{2a}, SO_{2v}, % – oxygen saturation in arterial (a) and venous (v) blood; pO_{2a}, pO_{2v} , *mm Hg* - partial pressure of oxygen in arterial (a) and venous (v) blood; a = 0,023 - a coefficient that depends on the temperature at which the blood oxygen content was measured, t=37°C; $Patm$ = 760 mm Hg - atmospheric pressure.
For the data processing were used the special algorithms of cluster analysis (Nastenko, 1996), correlation and regression analysis and variation statistics as well.
A detailed analysis of the dependencies and the corresponding regression equations are given in (Nastenko, e.a., 2000, 2001).

2.2 Empiric results

As the first step the interrelationships between systemic oxygen delivery (IDO_2) and consumption (IVO_2) (Nastenko, e.a., 2000, 2001) in intact cardiovascular system were studied (fig.1).

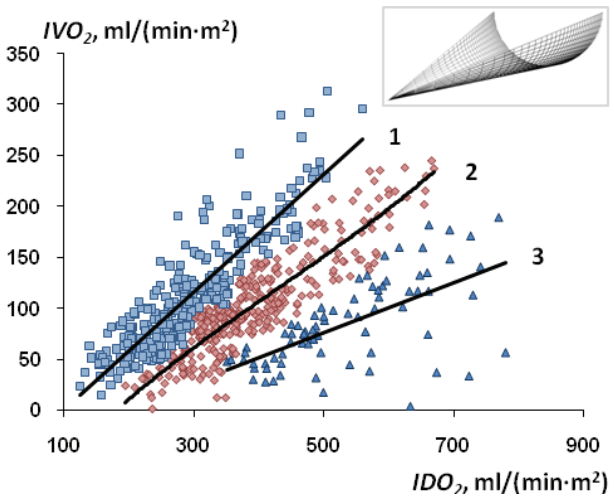


Fig. 1. Dependencies of systemic oxygen consumption (IVO_2) on its delivery (IDO_2).
1 – heart failure; 2 – normal regulation; 3 – heart hyperfunction.

With application of the special cluster analysis (Nastenko, 1996) we obtained the family of three dependencies $IVO_2(IDO_2)$ (fig.1). Their regression equations are presented below:

$$IVO_2 = 0{,}578 \cdot IDO_2 - 57{,}27;\ R = 0{,}89,\ N = 293;\ p<0{,}001; \quad (6)$$

$$IVO_2 = 0{,}456 \cdot IDO_2 - 75{,}90;\ R = 0{,}90,\ N = 317;\ p<0{,}001; \quad (7)$$

$$IVO_2 = 0{,}317 \cdot IDO_2 - 64{,}05;\ R = 0{,}89,\ N = 76;\ p<0{,}001. \quad (8)$$

We considered the mean values of system hemodynamic parameters and of gas content in arterial and venous blood for integrated estimation of parameters regulatory relations which generated the observing dependencies (Table 1).

The mean values of arterial (*MAP*) and central venous (*CVP*) pressures were approximately constant.

The differences of oxygen consumption (IVO_2) at the same oxygen delivery (IDO_2) can be explained by changes of ratio of nutritive and of shunting blood flow fractions.

Along every dependency IVO_2; IDO_2 were strongly correlated with *CI* (hydrodynamic opening of micro vessels at approximately the constant arterial pressure).

At the same time at fixed value of IDO_2 higher values of IVO_2 were observed at lower values of *CI* and correspondingly at higher systemic vascular resistance index (*SVRI*) (Table 1).

Low cardiac output causes the decrease of oxygen content in venous blood (Co_{2v}) at approximately constant oxygen content in arterial blood (Co_{2a}). Consequently the increase of

T №	Parameter	Number of dependency and observations quantity					
		(1) N = 293 obs.		(2) N = 317 obs.		(3) N = 76 obs.	
		M	$\pm m$	M	$\pm m$	M	$\pm m$
1	*HR*, min^{-1}	92,5	1,0	92,1	1,0	96,3*	1,9
2	*CI*, l/min/ m^2	1,77	0,03	2,26*	0,03	3,02*	0,09
3	*SVRI*, dyn·sec·sm^{-5}·m^2	3580	74	2708*	46	1996*	63
4	*MAP*, mm Hg	78,5	0,7	78,0	0,7	77,7	1,2
5	*CVP*, mm Hg	5,8	0,1	6,0	0,1	5,9	0,3
6	Co_{2a}, ml/dl	16,8	0,1	17,4*	0,1	17,9	0,2
7	Co_{2v}, ml/dl	10,6	0,1	13,1*	0,1	15,1*	0,1
8	$\Delta Co_{2a,v}$, ml/dl	6,2	0,1	4,4*	0,1	2,8*	0,1
9	PCO_{2a}, mm Hg	35,2	0,4	34,6	0,3	33,7	0,7
10	PCO_{2v}, mm Hg	42,3	0,4	40,6	0,4	39,2	0,8
11	*pHa*	7,42	0,02	7,43	0,01	7,44	0,02
12	*pHv*	7,36	0,02	7,37	0,02	7,38	0,02

*) the statistical difference in comparison with parameter for previous dependency is significant (p < 0,05…0,01).

Table 1. Mean values of system hemodynamic and biochemical parameters. Numbers of dependencies correspond to regression equations (6)-(8) and curves 1-3, fig.1

arterio-venous gradient of oxygen content ($\Delta Co_{2a,v}$) at low *CI* can be observed from two reasons: (i) activation of mass transfer through the capillary wall and (ii) from the activation

of oscillations frequency of capillaries (flickering of capillaries) with decrease of cardiac index (Knyshov e.a., 2009; Nastenko e.a., 2002) and predominant flow of blood through the zones with maximum oxygen debt.
Both of these mechanisms can be simulated with cellular automata models.

3. Cellular Automaton (CA) description

3.1 Theoretical preconditions of capillary network CA

Cardiovascular system (CVS) can be considered as interaction of combination of continual (arterial flow) and discrete (capillary flow) mechanisms (Chernukh & Alexandrov, 1984; Zveifach e.a., 974, 1977), which are studied insufficiently on present moment.
The relative regulatory autonomy of peripheral circulatory system can be simulated in more simple way by using of cellular automata models.
Changes in metabolic activity of tissues and the level of systemic blood flow causes to a change in the number of active capillaries and the duration of their active state. Each capillary is preceded by a sphincter, formed by several smooth muscle cells, which are an extension of the metarteriole muscular layer (Chernukh & Alexandrov, 1984; Zveifach e.a., 1974, 1977).
The block-scheme of single microcirculatory unit is shown in Fig. 2. It contains the real capillaries (intended for metabolic exchange) and arterio-venous (A-V) shunt microvessels.

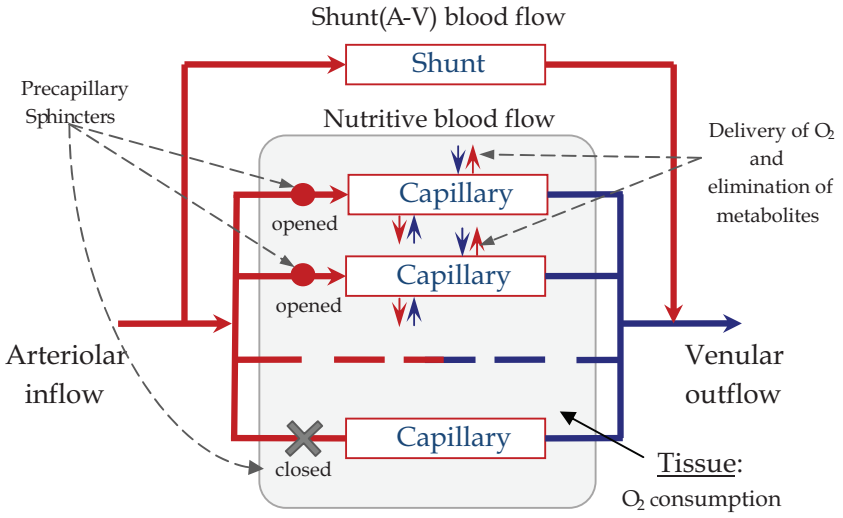


Fig. 2. Block-scheme of microcirculatory unit

At normal oxygen saturation of the surrounding tissues precapillary sphincter ***closes*** under the influence of the sympathetic nervous system (SNS), thus capillary blood flow stops (Fig.2).
The cells of tissue, adjacent to the closed capillary, produce the vazodilatory metabolites [1, 2, 4], which block the effect of the SNS. This leads to opening of sphincters, the resumption of blood flow through the capillary, oxygen flow and excretion of metabolites through the capillary wall. The action of SNS is restored after elimination of the accumulated metabolits.

The described principle of capillary functioning allowed us to represent the capillary as a discrete element with two states: "open" and "closed". This led to the idea of creating a model of capillary network in the form of a cellular automaton (CA).
Tissue oxygen saturation and the level of accumulation of tissue metabolites are inversely correlated. This allowed us to simulate the process of activation-deactivation of the capillaries, driving a single parameter - the "potential" of the capillary and tissue cells.

3.2 Decision rules of capillary network cellular automaton

The above thoughts allowed us to form the crucial principles of the microcirculatory network model in the form of cellular automaton containing two types of cells. Ones are simulating the tissue cells (carrying out metabolism) and others - with discrete states, simulating capillaries.
Consider the cross-sectional area Ω of tissues so that the central axis of the capillaries were perpendicular to the plane of the section. Idealizing the model, we assume that the central axis of the capillaries of the selected area of the tissue are parallel and located at equal distances from each other, unless otherwise indicated. To avoid edge effects we close the opposite edge of the area Ω. We divide the plane into square elements of the same size. Thus, in the grid nodes we have two types of elements: *tissue cells* - the elements that correspond to the cells of the tissue and *capillary cells* - the elements corresponding to capillaries.

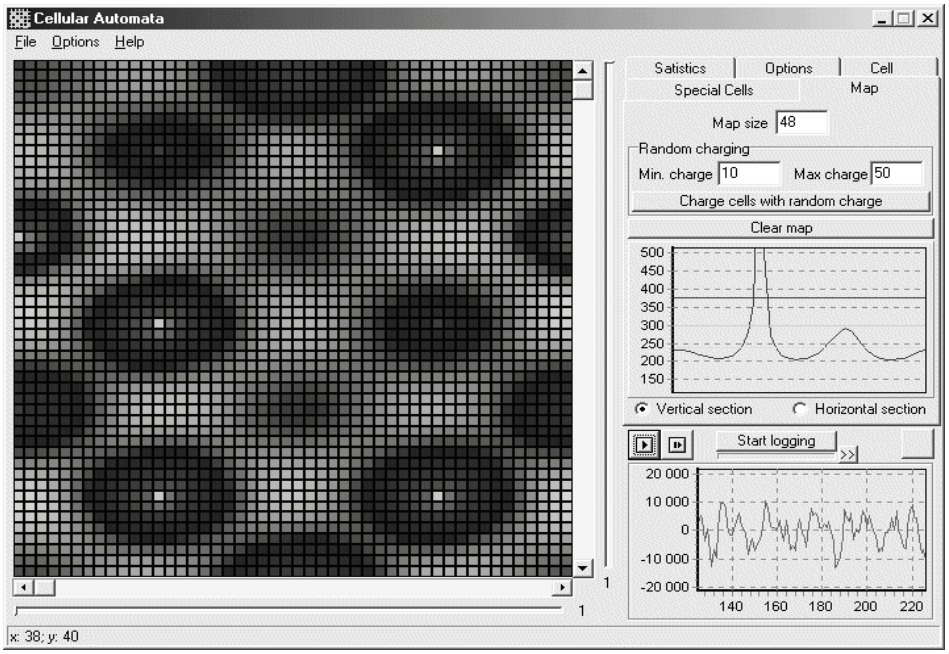


Fig. 3. The general view of cellular automaton.

Drawing an analogy with the physiological system, we introduce the quantity $z_{ij}^t \geq 0$ – the oxygen saturation of tissue cells at time t. For capillary cells we introduce the quantity $Z_0^t \geq 0$ - the concentration of oxygen in arterial blood.

We assume that the capillary in a closed state operates as a tissue cell, and that the concentration of oxygen in an open capillary remains constant and equal to Z_0. In more general case, $Z_0 = f(t)$ is a function.
In the following simplification we omit the existence of intercellular fluid and take a simplified scheme of the diffusion of oxygen directly into the tissue cells. Sequentially for each of the four cells we have an equation:

$$z_{ij}^{t+1} = 1/4\left\{z_{11}^{t} + z_{12}^{t} + z_{21}^{t} + z_{22}^{t}\right\}, \tag{7}$$

where z_{ij}^{t+1} - the oxygen content in tissue cells (or in a closed capillary) at time $t + 1$; z_{ij}^{t} - the oxygen content in tissue cells (or in a closed capillary) at time t.
Tissue cells consume oxygen, excreting metabolites. This process is described by equation:

$$z_{ij}^{t+1} = z_{ij}^{t} - a, \tag{8}$$

where a – is a value of the metabolic activity of cells, $a > 0$.
Abstracting from the multifaceted complexity of the biochemical processes responsible for opening and closing of capillaries, we introduce two quantities: the threshold of capillary opening, $\Theta_O > 0$, and the threshold of capillary closing, $\Theta_C > 0$, noting that $\Theta_O < \Theta_C$. Let's denote π - the region of sensitivity to influence of vasodilatory metabolites on precapillary sphincters. $Z_\pi = \sum_\pi z_{ij}$ - the total oxygen content in all tissue cells of π.
If for a closed capillary $Z_\pi < \Theta_O$, then the capillary is opened and blood oxygen saturation in it becomes Z_0 (arterial blood starts flowing through opened capillary). If for the opened capillary $Z_\pi > \Theta_C$, then the capillary is being closed and, as mentioned above, it starts to operating as a tissue cell.
The constructed model of microcirculatory network is a finite automaton. The state of matrix of cells changes abruptly, after the consequent application of decision rules to each cell of the matrix (within one iteration). Each iteration of the model contains following phases:

1. opening and closing of capillaries: for each capillary a value of Z_π is calculated; then it is being compared with thresholds Θ_O and Θ_C, and a new state of the capillary is being set according to algorithm described above;
2. the diffusion of oxygen in tissue cells (7);
3. oxygen uptake and excretion of metabolites, (8).

For simplicity of explanations, the range of values $a \in (0; a_{max}]$ is called the *regulatory range*. Value of a_{max} corresponds to a minimum value of the metabolic activity of tissue cells, which cease to switch the status of all capillaries. In this case, depending on other parameters of the model, all the capillaries or only some of them may be opened constantly.

3. Results of cellular automaton simulation

This section presents the qualitative properties of the integral microcirculatory network which do not depend on the conditions of the computational experiments and characterize the systemic behavior of microcirculatory network. All parameters are presented in percents of their maximal values, i.e., qualitative analysis of characteristics is allowed.
The investigations were conducted as follows. After setting the initial conditions, when the CA passed to a quasi-steady state, all necessary parameters were registered and analyzed.

Further, the intensity of tissue metabolic activity was changed and computational procedure was repeated.
Since the simulation pursued to study only the qualitative properties of microcirculatory network, so only relative or conventional quantitative parameters of the capillary network functioning were used.

3.1 Static characteristics of the capillary network

In the computational experiments we obtained integrational depencies of the following indices on the metabolic activity of tissue cells (Fig. 4): total capillary blood flow (i.e., the number of open capillaries), the average oxygen saturation of tissue cells, the number of capillaries, which are opened and closed during one iteration.

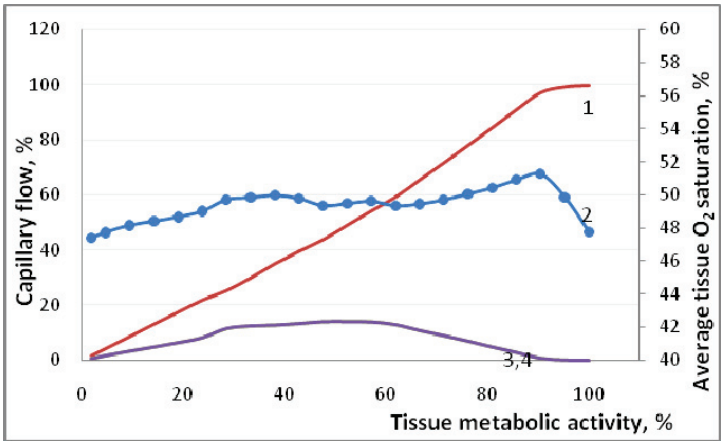


Fig. 4. Integral characteristics of capillary network dependent on tissue cells metabolic activity, represented in percents of maximum metabolic activity. 1 – capillary flow; 2 – tissue oxygen saturation; 3,4 – number of flickering capillaries (opened and closed during one iteration of change the state of all CA cells).

The number of open capillaries is linearly related to the metabolic activity of tissue cells (Fig. 3, curve 1). The total blood flow in the capillary network is equal to the amount of blood flow through active capillaries, whose number is determined by the current metabolic activity of tissue. Linear relation between capillary blood flow and metabolic activity of tissues is in good agreement with the linear dependence of oxygen delivery and consumption, obtained from the clinical data (Fig. 1).
The average value of oxygen saturation of tissue cells ($z_{ij,cp}$) varies slightly in the region (0,46; 0,54) in the range (0; 0,9 a_{max}) of metabolic activity (see Fig. 3, curve 2).This indicates the stability of the model to changes in metabolic activity of tissue cells.That is a real system, which has similar properties as represented model has, supports described characteristics without special homeostatic regulatory mechanisms.
The dependency of the average number of capillaries, which open or close during one iteration, from the metabolic activity of tissue cells (Fig. 3, curves 3,4) have a maximum in the vicinity of 0,5 a_{max}. Note that, although within a single iteration, the number of opened and closed capillaries can vary greatly, the average values of these indices are almost

identical. It should also be noted that for values $a > 0{,}5\, a_{max}$ the average number of capillaries, which changed their status, decrease due to the fact that some of the capillaries remain open permanently.
The results show high adaptability of microvascular network to changes of metabolic activity in individual organ or whole organism, which allows the homeostatic maintenance of vital balance of oxygen and metabolites in tissues.

3.2 Investigation of the influence of density of capillary network on the magnitude of the regulatory range of oxygen transport

Age development of a living organism is associated with varying level of tissue "capillarization" in different age periods (growing up, adulthood and aging). So, the number of capillaries per unit volume of tissue or per cross-sectional area can vary substantially. For example, aging is associated with the so-called reduction of the micro vascular network, often due to a sclerotic or other type of its arterial branches damage.
Therefore, it seems to be interesting to obtain the relation between density of capillary network (capillary/tissue cells ratio) and the value of the maximum possible tissue metabolic request which can be satisfied.
In other words, we studied the effect of the of capillary network density to the width of the regulatory range.

3.2.1 Computational experiments conditions

Cellular automata parameters were as following. It was used a matrix of 96x96 cells, closed in the torus, and three variants of the location of the capillaries among tissue cells (step on X x step on Y x pitch vertical displacement): *Option 1* - 6x2x3; *Option 2* - 12x4x6 and *Option 3* - 24x8x12.

3.2.2 Results of simulations

The relative difference of the opening and closing thresholds of the capillaries was 10% of their mean value. Other parameters were the same as previously used.
Tissue metabolic activity were increased step-by-step after each recording of parameters performed in quasi-steady state of CA.
Computation experiment was terminated when the average value of tissue cells saturation began to decrease, and all capillaries became opened.
Simulation results for three types of capillary network density shown in Fig.5. At sub-maximal values of tissues metabolic activities it is observed accelerated growth of capillary blood flow.
Curves had the shape of a swan or a half loop of hysteresis. Attention is drawn to a sharp narrowing of the range of possible metabolic activity at the most “rarefied” capillary network (curve 1, Fig. 5).
With high accuracy, the dependencies were approximated by power polynomials of grade 3-4, $R^2 > 0{,}99$. For each dependency, they are placed from up to down according to the numbering in Fig. 5.
The next question, which was subjected to the study, is as follows: how is the width of the range of possible dissipation depends on a ratio of the total numbers of capillary (N_{cap}) and tissue (N_{tis}) cells - N_{cap}/N_{tis}.

For experiment conditions described above, the correlation between the N_{cap}/N_{tis} ratio and maximum possible tissue metabolic activity is linear. Its regression equation is also represented on Fig.6.

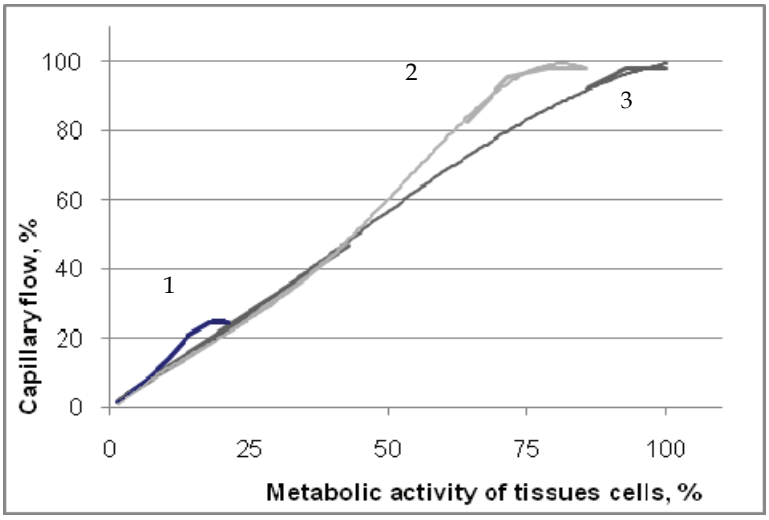


Fig. 5. Dependency of capillary blood flow on the magnitude of the metabolic activity for the three variants of the capillary network density: 1 - 6x2x3; 2 - 12x4x6; 3 - 24x8x12.

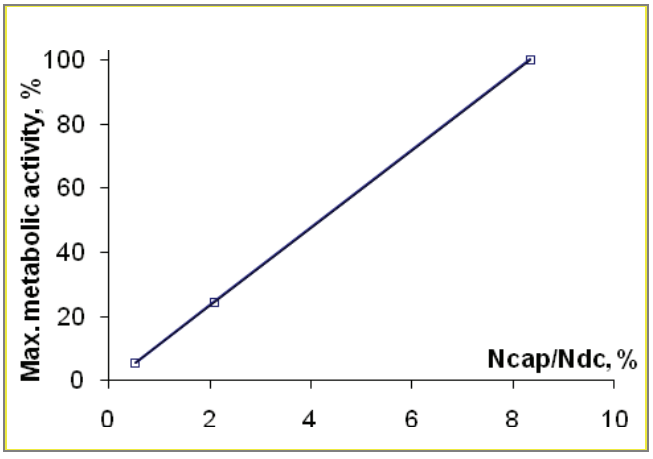


Fig. 6. Dependency of maximum possible tissue metabolic activity on the ratio of the capillary (N_{cap}) and dissipative (N_{tis}) cells - N_{cap}/N_{tis}.

Results of computational experiments suggest that at the age-related reduction of microcirculatory network, i.e. at reducing the tissue capillarization the metabolic request, which can be satisfied, decreases linearly. Conversely, when organism is growing up, the intensive development of microcirculatory networks can lead to an increase the width of the range of possible tissue metabolic activity. Apparently, this is one of explanation of the fact that the top sports results can be achieved at a young age. With ageing the physical ability of an organism decreases quite rapidly, due to decrease of tissues capillarization.
Similar arguments can be made with respect to coronary heart disease, when the rate of development of coronary arteries lesions determines the rate of progression of heart failure. Conversely, the intensity of revascularization after acute myocardial infarction may determine the degree of restoration of functional ability of the heart.

3.3 Dynamic properties of capillary blood flow and tissue oxygenation. Simulation results

Cellular automaton belongs to the critically self-organized systems (Bak e.a., 1987, 1996), with varying degrees of dynamic self-organization.
Depending on the computational experiment conditions, the capillary network demonstrates one of three types of functional self-organization:

- *Subcritical*: the capillaries are opened and closed at random time moment (self-organization is absent);
- *Critical*: joint regular and random distribution of open capillaries (the system goes from self-organized into a chaotic state and vice versa);
- *Supercritical*: open capillaries form a regular, lattice structures, in which the capillaries are or always open or rhythmically oscillate (the system is stable self-organized).

The forms of oscillations of capillary flow and tissue oxygen saturation obtained from CA simulations are presented on fig. 5. The wide range of its oscillation properties allows attributing it to the fourth class, characterized the most diversive, most complex behavior (Wolfram, 1984) and consequently most adaptive to changes of external and internal conditions.

3.4 Thresholds of opening and closing of capillaries

In available information resources we did not find any data about the level at which tissue oxygen saturation and at what concentration of tissue metabolites the activation and deactivation of the capillaries is occured, wheather these values are constant or variable. CA simulation allows to objectify the whole range of the capillary network behavior and to compare the results with the real capillary system functioning.
The **purpose** of this section was to study the influence of difference of opening and closing capillary thresholds on the formation of capillary blood flow oscillatory properties at different levels of metabolic activity of tissues.
It should be noted that there is no physical analog of the quantity "verge of discovery" or "closure threshold" of the capillary. Physically, these values represented by a combination of factors and various physical characteristics that affect opening and closing of precapillary sphincters. In general, these values may change in time. In ongoing studies, we identify only one value which determines the state of capillary (opened/closed) - the number of metabolites accumulated in the tissues in the neighborhood of exact capillary.

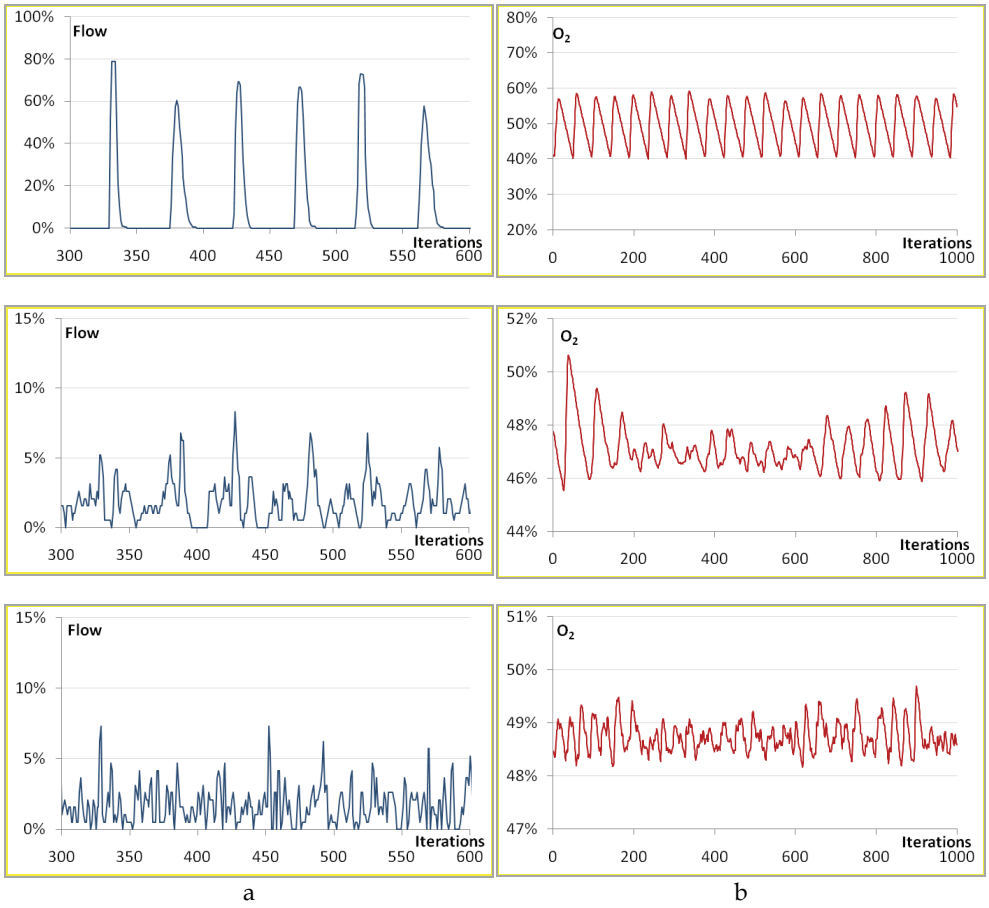


Fig. 7. The range of oscillation properties of capillary flow (a) and tissue oxygen saturation (b) obtained from CA simulations.

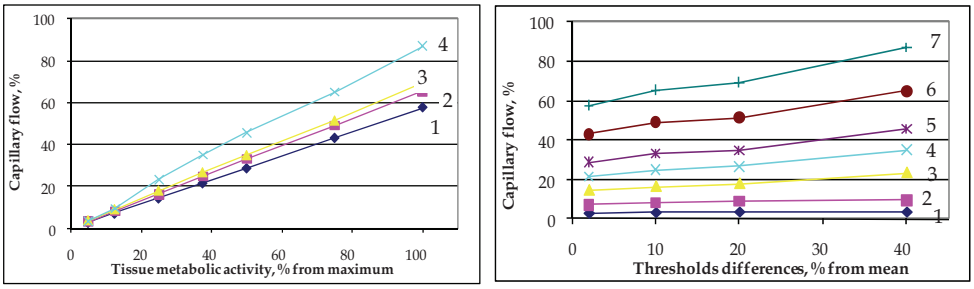
3.4.1 Simulation results

During the study we investigated the *static* properties of the CA - the average oxygen saturation of tissue cells, percentage of opened capillary cells (capillary blood flow); and *dynamic* properties as well - the percentage of capillary cells, which opened or closed during each iteration.

Tissues metabolic activity was calculated as a percentage of the maximum possible value. Differences of thresholds of capillary opening and closing were calculated as a percentage of their average value.

The average percentage of opened capillaries (of the total), depending on the tissue metabolic activity and the difference between the opening and closing of the thresholds presented in Fig. 8.

The average value of capillary blood flow is linearly dependent on tissue metabolic activity for all differences of capillary opening and closing thresholds. With increasing these difference the average number of opened capillaries increases by 20-38% at a constant metabolic activity of tissues.



a b

Fig. 8. The dependencies of capillary flow on: a - tissue metabolic activity; b - capillary opening and closing thresholds differece.

The average number of opened capillaries equal to average number of closed capillaries. Despite this a different number of capillaries opens and closes at a time (the difference reaches 25-30% of total), which causes additional fluctuations of blood flow in the capillary network.For these terms of computational experiment, the average number of opened capillaries is equivalent to the value of capillary blood flow.

For every difference of capillaries opening and closing thresholds the depencies of capillary flow on the tissue metabolic activity, as seen in Fig. 8,a are linear. The family of capillary flow curves formed by changes of capillaries opening and closing thresholds difference also was close to linear, Fig. 8,b. At the same metabolic activity of tissues, the capillary blood flow increases with increase of thresholds difference, Fig. 8,a. This can be explained by the longer time required to reach the necessary tissue oxygen saturation, that is more inert characteristics of the system.

The foregoing is confirmed by the family of curves Fig.1,b. At constant metabolic activity of tissues within each of the dependencies the capillary blood flow is than higher, than greater the difference between thresholds. This pattern is most clearly seen at high metabolic activity of tissues (Fig. 1,b, curves 1-5 from top to bottom) and almost not visible at low (Fig. 1b, two lowest dependencies).

It is also interesting the answer to the question, what is the percentage of flickering capillaries relative to the average number of active, in other words, what is the ratio of the dynamic (redistributing blood flow by opening and closing some part of capillaries) and the static components of the capillary blood flow. The results are presented on Fig.9.

For a small (up to 2% of the mean) difference of capillaries opening and closing thresholds the number of flickering capillaries is constantly equal to 100%. Some group of them opens and the same group is closing. This leads the most dynamic redistribution of blood flow to areas of tissue with the lowest oxygen saturation.

The number of flickering capillaries rapidly decreases with increasing of the threshold difference, Fig.9,b.

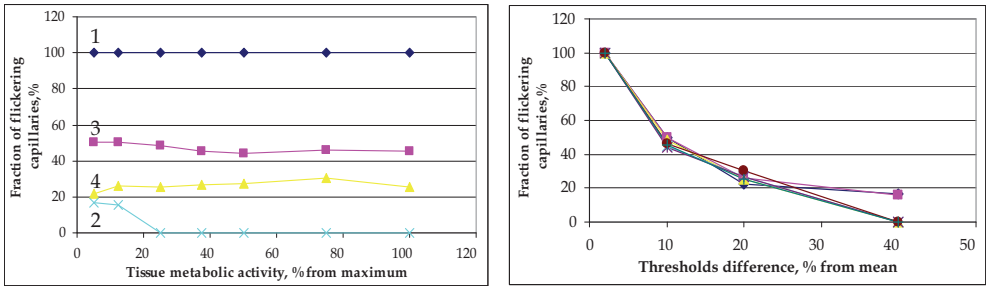


Fig. 9. Fraction of flickering capillaries, % on: a - tissue metabolic activity; b – opening and closing thresholds difference.

When the difference between the thresholds is equal to 10% on average of active, only the 50% of capillaries are opening and closing. With further increase of the thresholds difference the 22% and 16% of capillaries are flickering.
With increasing of the thresholds difference the number of flickering capillaries decreases in a hyperbolic law. Zero values indicate that the flickering stops and active capillaries become permanently opened.
Thus, the increase of thresholds difference may cause the deterioration of capillary flow redistribution.

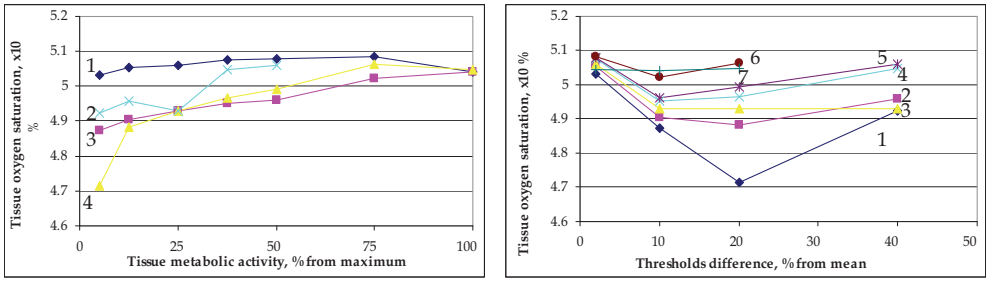


Fig. 10. The dependencies the average tissue oxygen saturation, % on: a - tissue metabolic activity; b - opening and closing thresholds difference.

The system "capillaries-tissue" is characterized by the following peculiarity, Fig.9,a. At a constant thresholds difference the number of flickering capillaries is constant, regardless of the metabolic activity of tissues.
For a small difference thresholds almost all capillaries are flickering. With increasing of the thresholds difference up to 10% of average number of active, the fraction of flickering capillaries varies between 42-55% of those active. With further increasing of the thresholds difference the number of flickering capillaries is reduced to zero. That is, the redistribution of nutritive (capillary) blood flow deteriorates with increasing of difference between the thresholds. This is confirmed by the schedules presented in Fig.9,b. Number of flickering capillaries, and, respectively, fraction of redistributed nutritive flow rapidly decreases with increasing of the thresholds difference.
It is observeed some analogy with acute heart failure, accompanied by tissue hypoxia. In order to saturate the tissues with oxygen and remove tissue metabolites, it is requires more

time in which the capillaries remain open. This situation may correspond to a large difference between thresholds of the opening and closing of capillaries.
Let's consider how the above-described functional peculiarities of capillary network influence on the tissues oxygen saturation, Fig.10,a. At the small thresholds difference the tissue oxygen saturation remains approximately constant in the entire range of metabolic activity of tissues. There is a tendency to some increase in the zone of medium metabolic activity with a gradual decrease in the area of high values of metabolic activity.
With increasing the difference of thresholds observes the tendency to increase the tissue oxygen saturation with increase of tissue metabolic activity. This dependency is even more expressed on higher magnitude of thresholds difference.
If the real capillary network has properties similar to those simulated in our investigations, for more severe conditions of organism, the conditions of oxygen delivery to tissues can be improved, due to the activation of the mechanisms of mass transfer, Fig.10,b.
This peculiarity was clearly expressed for low and medium levels of tissue metabolic activity and gradually decreases with increasing of this parameter.

6. Cellular automaton simulations of arterial and capillary flow interactions

Cardiovascular system (CVS) can be considering as interaction of combination of continual (arterial flow) and discrete (capillary flow) mechanisms (Chernukh & Alexandrov, 1984; Little, 1989; Rushmer, 1976) which interactions have been studied insufficiently at present moment.
The relative regulatory autonomy of peripheral circulatory system can be simulated in more simple way with use of cellular automata models.
We based our investigation assuming that microvascular arterial bed is critically self-organized system, working on the "edge of chaos". The microvascular arterial network increases or decreases diameter of arterioles and opens more or less capillaries depending on fluctuations of blood flow and pressure.
High sensitivity of capillary network to changes of internal and external conditions as well as a wide range of its oscillatory properties and also experimental data allows to assume that oscillations of peripheral and systemic blood flow can co-interact.
The purpose of this part of investigations was **t**o simulate interactions of capillary blood flow and systemic blood flow during the single cycle of heart contraction.

6.1 Conditions of simulation

1. Current level of systemic flow determines the quantity of capillaries, which can work simultaneously in this moment. But cellular automaton opens as much capillaries, as necessary to satisfy the oxygen debt, but not more than permitted.
2. A single iteration of CA equals to a time unit.
3. The quantity of possibly opened capillary cells is constant during diastole.
4. The tissue metabolic activity changes from 0% to 100% with discrete step for every numerical experiment.
5. CA opens capillary when average tissue oxygen saturation is lower than "opening threshold". The number of opened special cells is determined by conditions 1, 3.
6. Opened capillary cells closes if an average charge of surrounding tissue cells reaches the value of "closing threshold".
7. The properties of all capillary cells and tissue cells where homogenous throughout CA.

8. It was estimated the average oxygen saturation of tissue cells, the quantity of opened capillary cells (capillary flow), and the quantity of cells opened and closed on last iteration.
9. Form of oscillation of systemic flow was assumed to be constant in all computational experiments.
10. All parameters where recorded when CA was brought to stationary mode.

6.2 Results of simulations

Four types of interactions between systemic and capillary flow were found, fig.11:

a. *steady state*, low metabolic activity, capillary and systemic flows are not synchronized;
b. *normal arterial flow* – capillary and systemic flow are synchronized;
c. *essentially increased arterial flow* (for example, stress regulation) – the groups of capillaries open and close before the end of systole that may cause the hydrodynamic overload from the increase of local vascular resistance and can damage of endothelium of micro vessels;
d. *essentially decreased arterial flow*, tissue hypoxia: some capillaries do not close at the end of systole from the high concentration of vasodilatory metabolites that cause the decrease of diastolic arterial pressure.

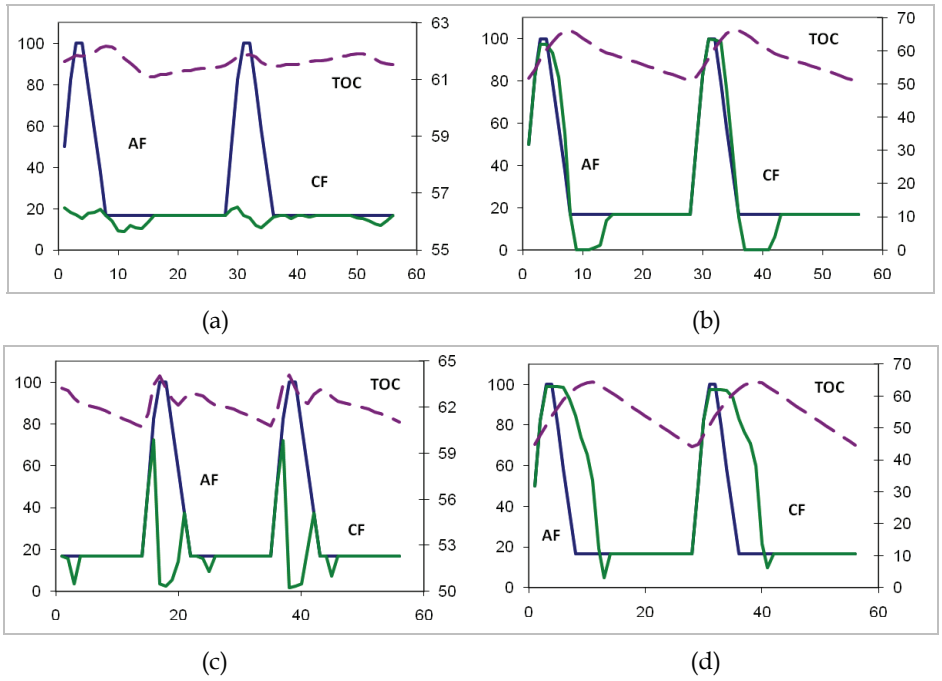


Fig. 11. Types of arterial (AF) and capillary (CF) flow interactions; oscillations of tissue oxygen consumption (TOC). Simulations with cellular automaton.

The experimental data, presented in (Lightfoot, 1974) confirm the presupposition about the presence of peripheral and systemic flow interactions. The oscillations of blood flow in arterioles and venulaes obtained from experimental data are caused by heart beating.

6.3 Simulation of atrial fibrillation conditions

On this step of study, we used the real patient data with atrial fibrillation, fig.12. The systemic flow supposed to be approximately proportional to blood pressure oscillations. Then we used the CA rules pointed in previous step.

In this case at essential variation of heart rhythm and cardiac output and at certain level of tissue metabolic activity the synchronization of arterial and capillary flow was possible, fig. 13.

The capillary flow changes synchronously with arterial. The situations of surplus blood flow at the state close to basal metabolism, Fig. 13a, and when the systemic flow is higher than necessary, Fig.13, b were similar to presented on fig. 11a,c.

Presented simulations allow to suppose that capillary system functioning can be synchronized with oscillations of arterial flow.

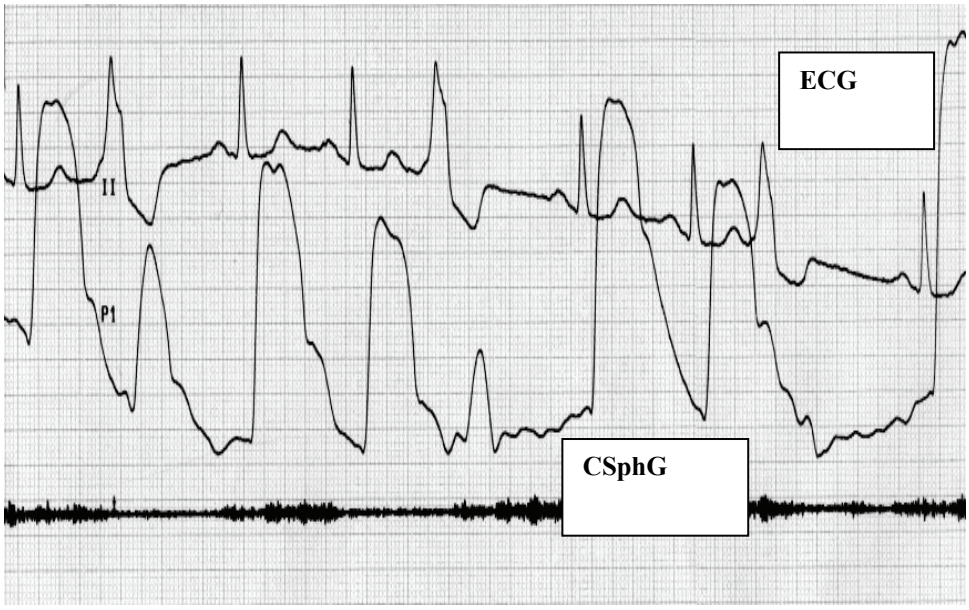
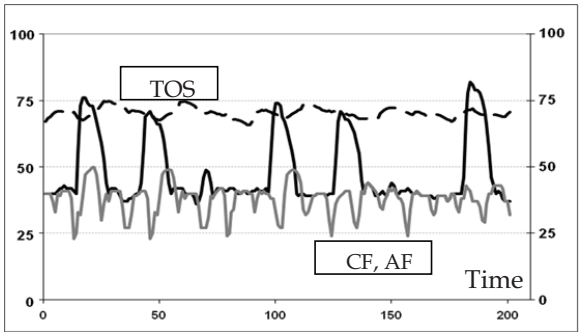
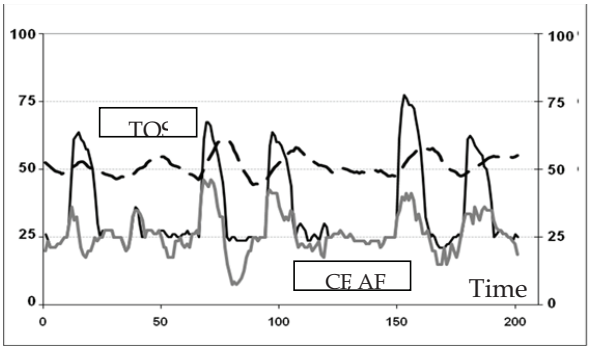


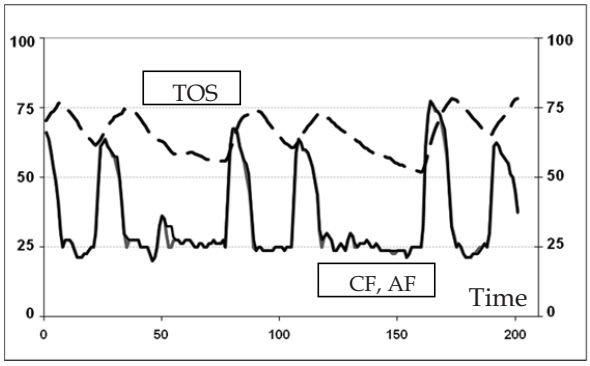
Fig. 12. Real Patient data for simulation of atrial fibrillation (Electropcardiogram – ECG and carotid shygmogram - CSphG).



a



b



c

Fig. 13. Synchronization of arterial and capillary flow from simulations, based on real patient data with atrial fibrillation. TOS – tissue oxygen saturation; AF – arterial flow; CF - capillary blood flow.

Different types of such interactions obtained from modeling can be a platform for planning of experimental investigations as well as for quantitative simulations of organism behavior.

7. Conclusions and future work

Cellular Automata Model of capillary blood flow is critically self-organized system and can be attributed to the automata of fourth class (Wolfram, 1984) that exhibit the maximum complexity and divercity of behavior. Carried out model experiments can extend our understanding of the systemic regulation of capillary blood flow and analyze a number of conditions that are difficult or impossible to obtain experimentally.

Carried out model experiments showed high adaptive properties of microcirculatory network to changes in internal and external conditions. A multiplicity of oscillatory properties of the capillary blood flow and the high complexity of behavior demonstrate the ability to synchronization the capillary and the systemic blood flow. At the same time the tissue oxygen saturation is maintained on physiologically acceptable level automatically, without any special regulatory mechanisms.

Two types of pathological interaction of capillary blood flow with systemic blood flow were obtained: for surplus systemic flow in comparison with systemic oxygen debt (stress, heart hyperfunction), as well as for insufficient blood flow (low cardiac output syndrome).

The obtained results allow understanding better the possiblities of vascular endothelium damage, which causes changes in the relations of arterial blood pressure parameters at different circulatory disorders, and on the basis of theoretical results to make the planning of the physiological and clinical studies.

In future we are plannig to use the obtained data for development of systems for clinical assessment of circulatory disorders as the violation of interactions of microcirculatory system and central hemodynamics, in particular – to estimate the pathological changes of blood pressure parameters, and on this basis to develop new approaches to the assessment of circulatory disorders and estimation of the effectiveness of therapy.

Cellural automaton models of microcirculatory system can be important component for systemic circulatory regulation modeling.

8. References

Achakri H., Rachev A., Stergiopulos N., Meister J.J. A theoretical investigation of low frequency diameter oscillations of muscular arteries. //Ann. Biomed. Eng. 1994. Vol. 22. N 3. P. 253-263.

Bak P. How nature works: the science of self-organized criticality. - Springer-Verlag New York, Inc. 1996. - 205 p.

Bak R., Tang C., Wiesenfeld K. Self-organized criticality: an explanation of 1/f noise //Phys. Rev. Lett. – 1987. Vol. 59. №4.-R 381-384.

Cavalcanti S., Ursino M. Chaotic oscillations in microvessel arterial networks. //Ann. Biomed. Eng. 1996. Vol. 24. N 1. P. 37-47.

Chernukh A.M., Alexandrov P.N. Microcyrculation.-Moscow.- Medicine, 1984.-432 p. (*Rus.*)

Kaplan J. Cardiac Anesthesia. // Philadelphia.-1979.-530 p.

Knyshov G., Nastenko E., Maksymenko V., Kravchuk A. Nutritive flow, shunt flow and peculiarities of microcirculation in regulation of oxygen transport // Proceed. of 10th Eur. Congr. on Extra-Corp. Circ. Technol. – Funchal, Portugal. – 2003. – P. 69-75.

Knyshov G., Nastenko Ye., Maksymenko V., Kravchuk O., Shardukova Yu. The Interactions between Arterial and Capillary Flow. Cellular Automaton Simulations of Qualitative Peculiarities.- WC 2009, IFMBE Proceedings 25/IV, - 2009. P. 572–574.// www.springerlink.com

Lightfoot E. N. Transport phenomena and living systems: biomedical aspects of momentum and mass transport. New York: John Wiley and Sons. 1974. 495 p.

Little R. C., Little W. C. Physiology of the Heart and Circulation. // Year Book Med. Publ. Inc.- 1989.-379 p.

Nastenko E., Maksymenko V., Belov Yu., Kravchuk A. Modeling of complex behaviour of the microvascular arterial network with cellular automata // Mathem. Modeling & Computing in Biology and Medicine. 5th ESMTB Conference 2002. – Ed. By V. Capasso. – MIRIAM. – Italy. – P. 227-234.

Nastenko E.A., Maksymenko V.B., Palec B.L., Onishchenko V.F., Rysin S.V. The role of central hemodynamics in the regulation of systemic oxygen transport // Yearbook of scientific works of the Association of Cardiovascular Surgeons of Ukraine - V. 8. - 2000. P. 142-144. (*Rus*.)

Nastenko E.A., Maksymenko V.B., Palec B.L., Rysin S.V. Investigation of the role of peripheral vascular resistance in the optimization of system of oxygen transport in norm and at heart failure.\\ Yearbook of scientific works of the Association of Cardiovascular Surgeons of Ukraine.-2001. -VOL.9 – C. 227-231. (*Rus*.)

Nastenko E.A. The Use of Cluster Analysis for Partitioning Mixtures of Multidimensional Functional Characteristics of Complex Byomedical Systems.- J. of Automation and Information Sciences.-V.28,-N5-6,-1996.- P.77-83.

Naylor-Shepherd M.F., Fuchs D.W., Angaran D.M. Oxygen homeostasis: theory, measurement, and therapeutic implications. // DISCP, Ann. of Pharmacotherapy.- V.24,- 1990. -P.1195-1203.

Ream A.K., Fogdall R.P. Acute cardiovascular management anesthesia and intensive care. // Philadelphia-Toronto: J.B. Lippincott Company.- 1982.-940 p.

Reeder G.D. The biochemistry and physiology of hemoglobin. // Am. Soc. of Extra-Corp. Technol., Inc., AMSECT.-Reston.- Virginia.-1986.- 250 p.

Risk Management: Risk. Sustainable development. Synergetics. - M.: - Science. - 2000.-431 p. (*Rus*.)

Rushmer R. F. Cardiovascular Dynamics, 4th edition, Philadelphia, W. B. Saunders Co.- 1976.-584 p.

Samsel R.W, Shumacker P.T. Oxygen delivery to tissues.//Eur. Respir.J., 4,-1981. P.1258-1267.

Wolfram S. Universality and complexity in cellular automata // Physica D. Vol.1. 1984. P. 91-125.

Wolfram S. Computation Theory of Cellular Automata. In: Theory and Application of Cellular Automata. World Scientific: Singapore. 1984. P. 189-230.

Yusupov R.M., Polonnikov R.I. Telemedicine - the new information technologies on the threshold of the XXI century.-St .- 1998.-490 p. (*Rus*.)

Zweifach B.W. Quantitative studies of microcirculatory structure and function III Analysis of pressure distribution in the terminal vascular bed in cat mesentery // Circulation. – Res. – 1974. – V. 34. – P. 843-857.

Zweifach B.W., Lipowsky H.H. Quantitative studies of microcirculatory structure and function. III. Microvascular hemodynamics of cat mesentery and rabbit omentum// Circulation Research, -V. 41.- 1977. – P. 380-390.

Part 3

Dynamics of Social and Economic Systems

15

Social Simulation Based on Cellular Automata: Modeling Language Shifts

Francesc S. Beltran[1], Salvador Herrando[1], Violant Estreder[2], Doris Ferreres[2], Marc-Antoni Adell[2] and Marcos Ruiz-Soler[3]
[1]*Universitat de Barcelona,*
[2]*Universitat de València*
[3]*Universidad de Málaga*
Spain

1. Introduction

Nowadays, language shifts (i.e., a community of speakers stops using their traditional language and speaks a new one in all communication settings) may produce a massive extinction of languages throughout the world. In this context, an important task for social sciences research should therefore be to achieve a deep comprehension of language shifts. However, modeling the social and behavioral variables that guide the social behavior of individuals and groups has traditionally been tricky in all the social sciences. In this situation, social simulation provides a tool for testing hypotheses and building models of social phenomena (see, for example, Gilbert, 1996; Gilbert & Toitzsch, 2005; and Goldspink, 2002), especially the techniques based on cellular automata theory (Hegselmann, 1996; Hegselman & Flache, 1998; Nowak & Lewenstein, 1996). According to this framewok, we introduce the properties of a cellular automaton that incorporates some assumptions from the Gaelic-Arvanitika model of language shifts (Sasse, 1992) and the findings on the dynamics of social impacts in the field of social psychology (Latané, 1981; Nowak et al. 1990). Thus, we define a cellular automaton and carry out a set of simulations in which it is used. We incorporate empirical data from recent sociolinguistic studies in Catalonia (a region in Southern Europe) to run the automaton under different scenarios. The results allow us to highlight some of the main factors involved in a language shift. Finally, we also discuss how the social simulation based on cellular automata theory approach proves to be a useful tool for understanding language shifts.

2. A sociolinguistic model of language shifts

Although there are languages spoken in the past that are not spoken today, e.g., Etruscan, Egyptian and Hittite, and people usually refer to them as dead languages, the death of a language is not only an ancient event. UNESCO (2003) estimated that, by the end of this century, more than 5,100 of the approximately 6,000 languages currently spoken around the world will have disappeared; i.e., approximatelly 90% of them. When a language dies, the community of people that speak that language lose a main element of their identity and their cultural framework is impoverished as a result. The most likely future of that

community is its assimilation into a larger cultural group. Hence, the death of a language implies an irreversible impoverishment of the world's cultural diversity. In summary, language death is a major cultural problem today because (a) the large number of languages affected by extinction includes several million people and (b) humankind's cultural wealth is reduced as a result of language extinction.

Why does a language die? Obviously a language dies if its speakers disappear, either due to an action such as direct genocide or genocide through the destruction of their habitat or economic resources. But usually a language dies because the speakers *decide* to abandon the traditional language and to adopt a new one in all communication settings (Mühlhäusler, 1996). Note that the key factor for declaring that a given language becomes extinct is usage, not the linguistic competence of the speakers. So, the next question is what factors impel a whole community of speakers to shift from one language to another? One premise is that such community must be fluent in at least two languages. Then, if there are two or more languages in a community, a hierarchical structure is frequently adopted, with one becoming the dominant language (DL) and the other the subordinate language (SL). Althouhg it is possible for both languages to coexist within such a hierarchy for long periods of time, historical events can disturb the equilibrium. In these cases, the speakers of the SL may notice that their language has lost value relative to the DL. They may then *decide* that it is no longer useful and stop speaking it in all domains of use. Hence, there are three phenomena involved in language death (Sasse, 1992): (a) the cultural, historical, sociological and/or economic factors which create pressure to abandon the language in the speakers' community (the so called *external setting*), (b) the domains of use and the attitudes towards the languages of the speakers (so called *speech behavior*) and (c) the linguistic impoverishment observed in the morphology, phonology, syntax, etc., of the SL (so called *structural consequences*).

Although these three phenomena are interrelated (the pressure on the community created by the external setting compels speakers to modify their speech behavior, which produces an impoverishment of the structure of the SL), in the present study we will focus on the speech behavior of the individuals. Given the fact that an important issue related to language death is the language policies designed to reverse the language shift of threatened languages (Fishman, 1991), it is very important to take steps in the external setting where the language shift process occurs, i.e., by implementing government initiatives to ensure that the use of the SL is not mitigated. However, deciding to shift language or not is an individual decision made by each SL speaker. Therefore, it is also necessary to focus on individual factors relating to speech behavior to better understand a language shift and to design policies addressed to reverse language shifts.

Based on studies of the death of two languages in Europe, namely a variety of Scottish Gaelic and an Albanian dialect spoken in Greece, Sasse (1992) introduced the Gaelic-Arvanitika model. This model stated that one of the main factors involved in maintaining a language across generations is transmission within the family. If the parents speak to their children in a language other than their own, the language shift process will be completed in approximately two generations (see Figure 1). Although the Gaelic-Arvanitika model is biased towards an European context, it points out some relevant features involved in language shifts. For example, the death of a language is not a slow process lasting several centuries, but a fast process that can take a few decades.

Given the importance of attitudes towards the SL language in determining the speech behavior of the speakers, it is also necessary to take into account how individuals change their attitudes. In the field of social psychology, Latané (1981, 1996) explained how the opinions of

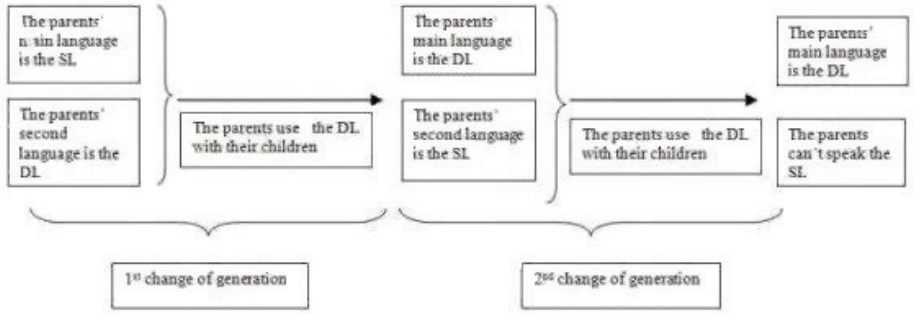


Fig. 1. Consequences of the interruption of language transmission in the family according to the Gaelic-Arvanitika model of language shifts: Given a dominant language and a subordinate language in a speaker community, the non-transmitted language (the SL) becomes extinct after two generations (from Beltran et al., 2009).

individuals change based on the social influence of the group they are in. According to Latané, the impact or social influence of a group over an individual is a product of three factors: (a) the strength over the individual, (b) the physical immediacy of group members and (c) the number of group members influencing the individual. The predictions of the theory and the dynamics of the social impact were studied exhaustively using both empirical research and simulation techniques (Latané et al., 1994, 1995; Latané & Wolf, 1981; Nowak et al., 1990). Similarly, we propose that an individual's speech behavior can be subjected to the same rules as social impact. We therefore hypothesize that a given individual will shift from the SL to the DL if he or she receives strong pressure from the individuals in the group and a considerable number of close neighbors maintain this pressure.

3. A language shift simulation based on cellular automata

3.1 A model of language shift based on cellular automata

There are currently many examples of potential language shifts around the world, so the social and cultural contexts where language shifts occur tend to vary. We developed a model involving a social context where two languages coexist and one is threatened with potential extinction. Our model states that the individuals will change their speech behavior in regard to the SL if they are weakly engaged with it and/or a considerable number of their neighbors maintain a different speech behavior. We can summarize the main features of speech behavior in our model as follows:

- It is a local behavior in time and space, because the decision to shift languages affects one individual at a given time.
- It is an autonomous behavior, because the external setting puts pressure on each individual to make the decision to shift languages, but this shift occurs without an explicit consensus with the members of the speaker community.
- It is mass behavior, because a great number of individuals make the decision to stop using their usual language and use the DL.
- It is parallel behavior, because the individuals make the decision to stop using their usual language and use the DL at approximately the same time.

All these properties produce a self-organized emergent social phenomenon because there is no centralized unit guiding the process and the overall result, i.e., the extinction of a

language, is not explicit in individual behavior. Note that the external setting that triggers a shift from a SL to a DL is usually a process guided by the group of DL speakers, which puts pressure on the speakers of the SL, but the language shift itself is an autonomous individual *decision* made by the speakers of the SL.

The behavior of the cellular automata exhibits properties of localism, parallelism, emergence, etc., as occurs empirically during a language shift. Thus, the transition rules of a given cellular automaton are frequently simple, but it is only possible to know the state of the cells in a given future time $t+k$ by running the automaton from $t=0$ to $t= k$. Similarly, it is possible to assume that the language shift is regulated by a set of simple rules at the local level (the speech behavior of individuals) which produces global behavior at the social level (the extinction of a language). If it is possible to define the transition rules that describe the main features of a language shift, running the automaton will make it possible to predict the future of a SL given different scenarios in the present.

According to our model, depending on the attitude towards the SL (i.e., the strength or weakness of individuals' engagement with the SL), the social pressure favoring the use of the DL and the number of neighbors engaged with the DL, the speech behavior of each person can be categorized in one of three main states. Each state number indicates the level of engagement with the SL, from zero (0) to maximum strength (2):

a. State 0: The person only speaks the DL.
b. State 1: The person usually speaks the DL, but also speaks the SL, depending on the communication setting. The person transmits the DL to his or her children.
c. State 2: The person usually speaks the SL, but also speaks the DL, depending on the communication setting. The person transmits the SL to his or her children.

Because of the hierarchical structure of the two languages, everyone usually knows the DL, but only a percentage of people know the SL. So a percentage of people are monolingual in the DL, but there are no monolinguals in the SL. To include the information about the speech behavior of individuals provided by the Gaelic-Arvanitika model, the definitions of states 1 and 2 include transmission of the DL or the SL to the next generation. Obviously, the speakers in state 0 transmit the DL to their children. The bilinguals transmit their preferred language to the next generation (the state-1 bilinguals transmit the DL and the state-2 bilinguals transmit the SL).

The speaker community of our model lives in a discrete two-dimensional torus-shaped world. The world contains 105x64 cells, with each cell containing an individual. In general, a simulation based on cellular automata makes use of an unlimited world (i.e., a torus) rather than a limited world (e.g., a square), because in a limited world the cells near the edge have incomplete neighborhoods. Moreover, a torus space in a language-shift simulation also shows that all individuals interact with each other without restriction. The amount data provided by the 6,720 cells makes it possible to do both statistical descriptions and visual analysis on the computer screen. At each unit of time, a cell can only be classified in one of the three possible language states (0, 1 or 2), indicating the individual's strength in the use of the SL. Our cellular automaton does not include the *birth* or *death* of cells, but each cell inherits the transmitted language when the generation is renewed.

A factor in determining the use of a given language is the number of interactions where it is possible to use that language. This includes *the submission rule*, a typical behavior of state-2 speakers, who tend to use the DL automatically when they address a DL speaker, even if the DL speaker is competent in the SL (for a complete explanation of the submission rule, mathematical modeling and language shift effects, see Melià, 2004). Thus, the number of

neighbors in each linguistic state also determines a given individual's use of the DL or the SL. In our model each cell has eight adjacent neighbors on the side and at the vertex (a Moore neighborhood with a radius of 1), and the sum of neighbor values indicates the social pressure on the individual to use the DL or the SL (a value between 0, if all cells in the neighborhood are classified in state 0, and 18, if all cells are classified in state 2). A low sum value means an individual has few opportunities to interact with his/her neighbors using the SL, but if the sum value increases, the individual's opportunities to interact using the SL also increase.

The transition rule determines the future state in time *t+1* of a given cell, which has a given state in time *t*. The new state of a cell depends on whether or not the sum of the neighborhood values, including the cell target, surpasses a previously defined threshold. There are three thresholds:

a. S_a: a sum value below the threshold produces a sharp transition, i.e., state 2 changes sharply to state 0.
b. S_b: a sum value below the threshold produces a transition from a higher-value state to a lower-value state, but a sum value above the threshold produces a transition from a lower-value state to a higher-value state.
c. S_c: a sum value above the threshold produces a transition from a lower-value state to a higher-value state.

		To state:		
		0	1	2
From state:	0	$\Sigma \leq S_b$	$\Sigma > S_b$	----
	1	$\Sigma < S_b$	$S_b \leq \Sigma \leq S_c$	$\Sigma > S_c$
	2	$\Sigma \leq S_a$	$S_a < \Sigma < S_b$	$\Sigma \geq S_b$

Table 1. The transition rule of the cellular automaton that simulates language shifts. Note that the transition from state 0 to state 2 is difficult to observe empirically, because it involves a monolingual speaker becoming bilingual with a preference for the SL.

The threshold values ($S_a < S_b < S_c$) indicate the individual's level of engagement with the SL. When there is a greater level of engagement, the individual needs a lower threshold value to move up to a higher-value state. So the individual increases his/her usage and transmission of the SL eventually increases with only a minimal number of current neighbors using the SL. Conversely, when there is a lower level of engagement, the individual needs a higher threshold value to move up to a higher-value state. So the individual decreases his/her usage and transmission of the SL eventually decreases if there is not a large number of current neighbors using the SL. The transition rule and an example are described in detail in Table 1 and Figure 2.

Our cellular automaton's universe, the states and the transition rule to simulate a language shift were implemented on a Microsoft® Excel spreadsheet. We defined three spreadsheets in an Excel book. One spreadsheet allowed the user to define the number of cells classified in each state at *t=0*, the threshold values (S_a, S_b and S_c) and the number of simulations, given an initial number of states and threshold values. The number of cells classified in each state was determined by indicating the probability of each cell falling into one of the three states at *t=0*. Another spreadsheet showed the cells and their states at each time unit. The state of the cell was indicated by a color: white for state 0, orange for state 1 and green for state 2. This spreadsheet also displayed the frequency of the states at each time unit. Finally, a third

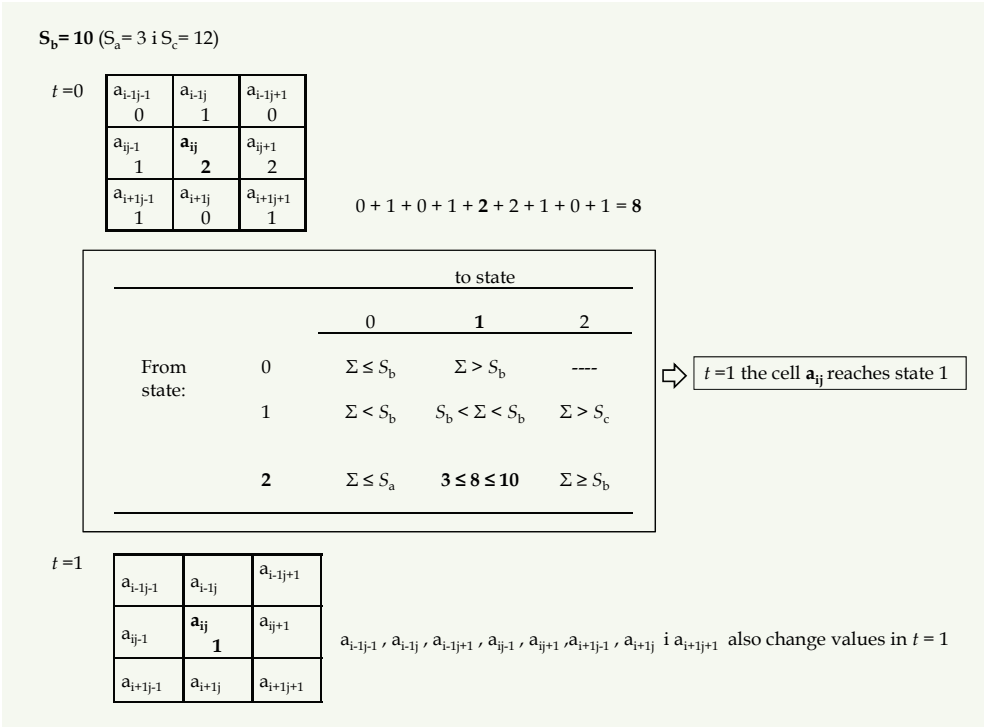


Fig. 2. An example of the transition rule in the cellular automaton that simulates language shifts. Given the thresholds equal to S_a= 3, S_b= 10 and S_c= 12, and the sum of the Moore neighborhood of radius 1 of the target cell equals eight, if the target cell is in state 2 at t=0, the transition rule states that the cell target will be in state 1 at *t=0* (the value of the sum of the neighborhood is between 3 and 10, the values of the thresholds S_b and S_c, respectively).

spreadsheet summarized the frequency of states for each simulation at each time unit until the automaton stabilized. Although the automaton runs automatically when the number of simulations is defined and the data are displayed on the spreadsheet where the frequency of states was indicated, the automaton can also be run step by step and display the evolution over time of the states on the spreadsheet that shows the cells and their states by color. (The Excel macros used to define the automaton and the main instructions to run it can be downloaded from www.ub.edu/gcai. Go to *download* in the main menu)

3.2 Testing the model: the example of Catalan

The availability of empirical data from recent language surveys on the use of Catalan prompted us to choose Catalan as an empirical example with which to evaluate our model. Catalan is a Romance language currently spoken by approximately ten million people along the Mediterranean coast from near Southern Spain to the South of France, the Balearic Islands and the town of Alghero in Sardinia (see Figure 3). This area is currently divided politically into four countries: Andorra, France, Italy and Spain, each of which grants a different official status to Catalan. Thus, Andorra recognizes Catalan as its single official

language, Spain recognizes Catalan as a joint official language in the regions where Catalan is spoken, and France and Italy do not grant Catalan any official status. Hence, the knowledge and use of Catalan varies accros the area where it is spoken and interacts with differents languages, such as French, Italian and Spanish.

In previous studies we tested the cellular automaton using some data from a language survey on knowledge and use of Catalan in Valencia, a region of Spain where Catalan is spoken (Ninyoles, 2005). The results of the simulations showed the automaton's extreme sensitivity to variations in threshold S_b compared with variations in thresholds S_a and S_c. Moreover, our simulations showed that, given the initial size of the current speech behavior of the individuals indicated by the cellular automaton states, the value of threshold S_b became critical in explaining the dynamics observed in the simulation. Thus, the results of our previous research stated that given a linguistic setting with an initial size of the current speech behavior of the individuals (Catalan and Spanish speakers in our research), the individual's social support for the SL, i.e., Catalan, becomes critical when determining the individual's speech behavior with regard to the SL (Beltran et al., 2009, 2010).

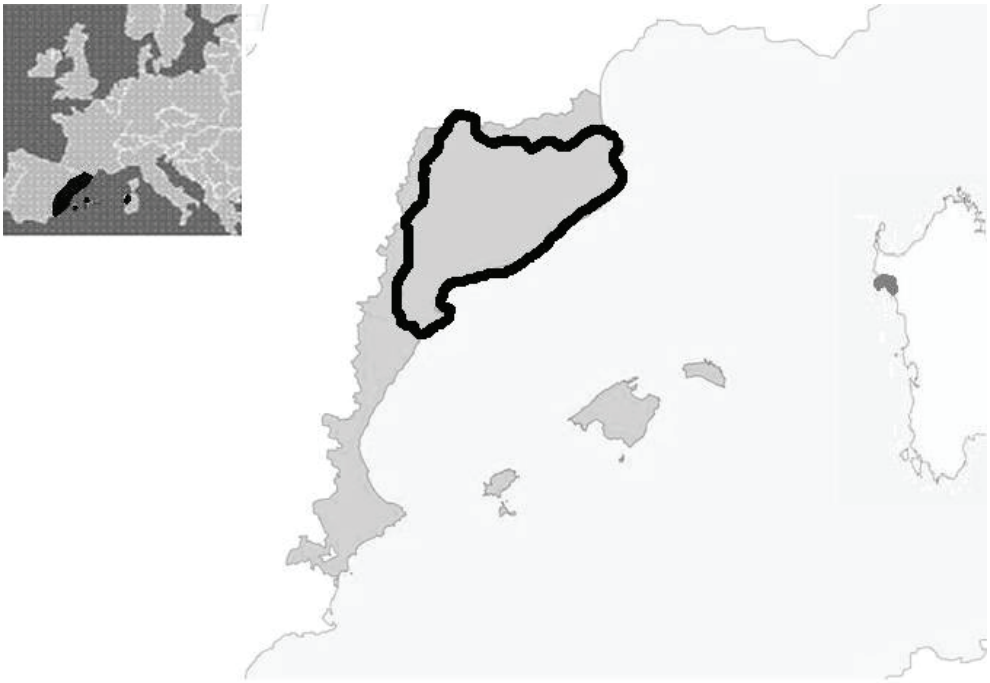


Fig. 3. Complete linguistic area where Catalan is spoken (grey area). The outlined area in black indicates Catalonia, the zona where we obtained the data from the linguistic survey used in this research study (Idescat, 2008). The map in the top-left corner shows the location of the whole linguistic area of Catalan in Europe.

In this study we used empirical data from a recent language survey carried out by the *Secretaria General de Política Lingüística* (Secretariat General for Language Policy) (Idescat, 2008) of the autonomous government of Catalonia, another region of Spain where Catalan is

spoken. The data from the survey were collected in 2008 from a sample of 7,140 people aged 15 and over on the use of Catalan with reference to different variables such as age, gender, educational level, place of residence, etc. We obtained data from the survey in a number of different social contexts and we systematically tested the effective use of Catalan in four social contexts: at home, with friends, in traditional stores and at large shopping malls. (These survey data are summarized in Table 2)

The percentages for the use of Catalan obtained in the survey gave us the number of cells containing each state at the beginning of the simulation (*t=0*). Thus, "Always Spanish" and "More Spanish than Catalan" were assigned to state 0; "Equal Catalan and Spanish" was assigned to state 1; and "More Catalan than Spanish" and "Always Catalan" were assigned to state 2. The percentage of states at *t=0* obtained after the conversion is shown in Table 3.

	Always Catalan	More Catalan than Spanish	Equal Catalan Spanish	More Spanish than Catalan	Always Spanish	Other language / Did not answer
Social context						
Home	31.6	3.6	8.3	6.0	42.6	7.9
Friends	22.5	10.8	16.8	9.0	33.9	7.1
Traditional stores	28.7	11.0	14.9	7.5	36.0	1.9
Large shopping malls	23.9	9.9	15.5	9.5	39.4	1.8

Table 2. Percentage of Catalan use in four social contexts. Data were obtained in 2008 from a sample of 7,140 people aged 15 years and over (Idescat, 2008).

State of the automaton	Home	Friends	Traditional stores	Large shopping malls
State 2	35.2	33.3	39.7	33.8
State 1	14.3	25.8	22.4	25.0
State 0	42.6	33.9	36.0	39.4

Table 3. Percentage of the states at *t=0* after conversion to the automaton states based on the linguistic survey data on Catalan use in four social contexts (Idescat, 2008).

Given the initial values, the variation in the threshold values gave us different scenarios of possible social support for the individuals using Catalan. Thresholds S_a and S_c were set at 3 and 15, respectively. As stated above, the results of previous simulations showed the automaton's extreme sensitivity to variations in threshold S_b compared with variations in thresholds S_a and S_c. The values of thresholds S_a and S_c were therefore kept constant in all simulations and threshold S_b was varied across seven values (4 to 10) in the four social contexts. The factorial combination of the four social contexts (at home, with friends, at traditional stores and at large shopping malls) and the seven thresholds S_b (4 to 10) produced twenty-eight different simulation conditions. We carried out 200 simulations for

each condition. The states were randomly seeded at *t=0* in all simulations, given that Catalan and Spanish speakers in Catalonia were very mixed. Random seeding therefore indicated the spatial distribution of the different kinds of speakers in our empirical example.

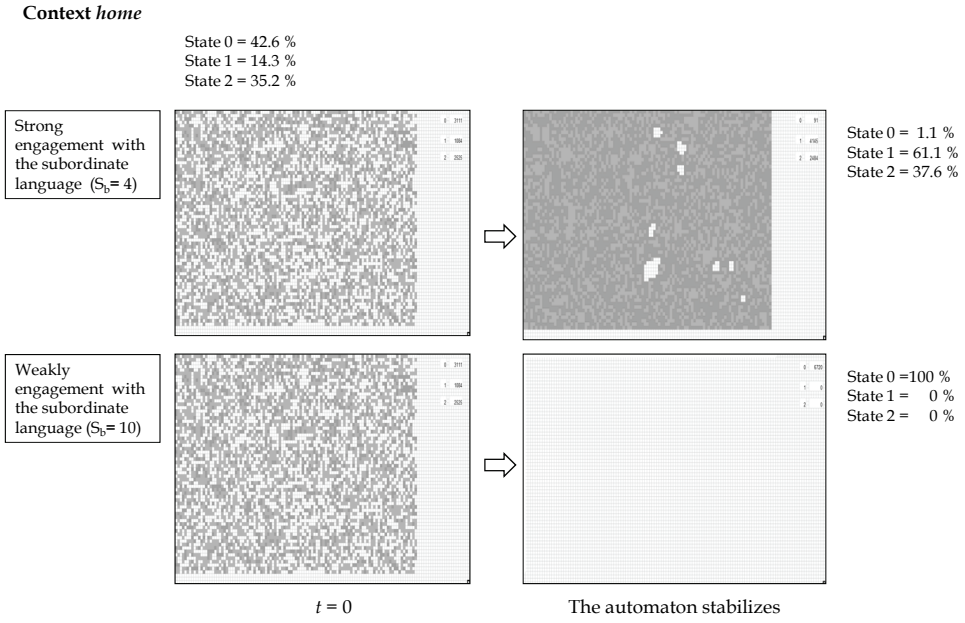


Fig. 4. An example of the dynamics of the cellular automaton that simulates language shifts. The initial values of the cellular automaton (*t=0*) were the percentage of states of the use of Catalan at home according to the linguistic survey (Idescat, 2008). The dark grey cells indicate state 2, the light grey cells indicate state 1 and the white cells indicate state 0 (on the Excel spreadsheet the states were represented by white, orange and green, respectively). Strong engagement of individuals with the subordinate language (threshold S_b= 4) allows the subordinate language to survive, because, when the cellular automaton stabilizes, there are bilinguals that use the subordinate language in states 1 and 2; but weak engagement (threshold S_b= 10) produces the extinction of the subordinate language, because, when the automaton stabilizes, all individuals become monolinguals in the dominant language (state 0).

We carried out the simulations for each condition and recorded the frequency of each state when the cellular automaton stabilized. The criterion of stabilization was three sequential iterations without changes in any state of all the cells of the automaton. The mean percentage of the 200 simulations was obtained for each state in each condition. The results coincide with the results obtained from previous simulations (see Beltran et al., 2010) and showed that below a given S_b threshold, state 0 disappeared and the SL survived, but above a certain S_b threshold, states 1 and 2 disappeared and the SL consequently became extinct (see Figure 5). In summary, the results suggested that, given the values of the states at t = 0 and a value of threshold S_b, we can determine whether the SL will survive or become extinct.

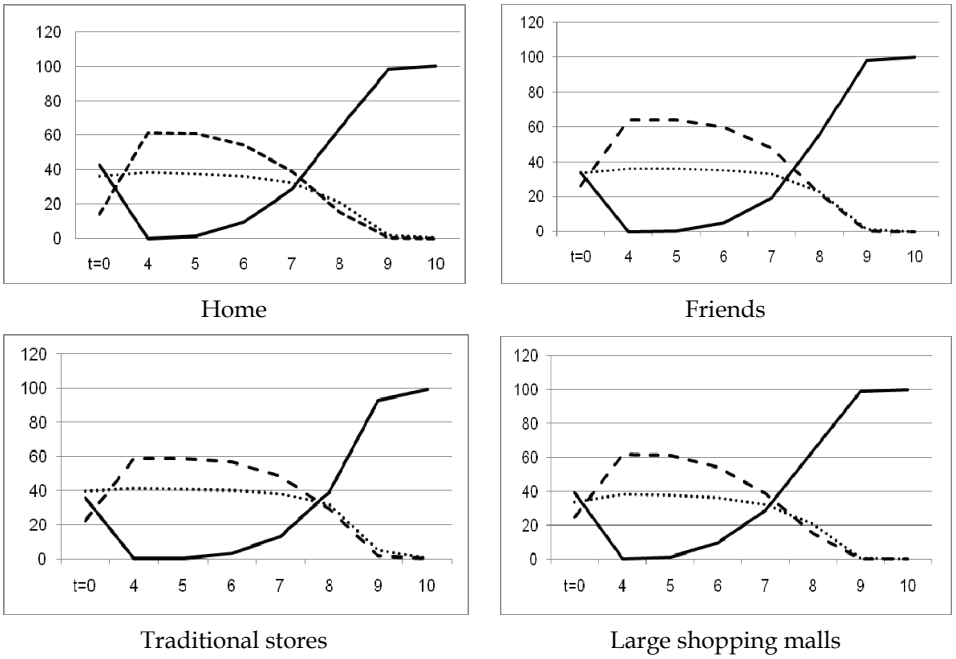


Fig. 5. Mean percentages of states 0 (solid), 1 (dashed) and 2 (dotted) for the values of threshold S_b when the automaton stabilized (values of S_b= 4 to 10) for each social context. The percentage of initial values (*t=0*) is also shown.

Moreover, as pointed out above, the Gaelic-Arvanitika model states that the language shift happens over a short period of time, namely two generations. We performed a second set of simulations to test that statement by determining whether the cellular automaton could forecast the progress or reversal of the language shift across generations. The procedure to simulate a change of a generation was as follows: We ran a given simulation and the data recorded after stabilization of the automaton were used as the initial values (*t=0*) of a new simulation, and so on. So each simulation run in the automaton indicates a change produced in a generation. As in the first set of simulations, the percentages for the use of Catalan obtained in the linguistic survey gave us the number of cells containing each state at the beginning of the simulation (*t=0*) (Idescat, 2008). In this second set of simulations, we chose only the context *home*, because the Gaelic-Arvanitika model stated that the key factor for the survival of a language is transmission in the family across generations.

Thresholds S_a and S_c were set at 3 and 15, respectively, and threshold S_b was varied across six values (4 to 9). According to the procedure mentioned above, the results of a given simulation furnish the initial values of a new simulation. This procedure was carried out four times for each value of threshold S_b and 15 simulations were performed for each value of S_b. The frequency of each state was recorded when the cellular automaton stabilized at the end of each simulation and the mean percentage of each state was obtained from the records of the 15 simulations. Thus, for each generation the mean percentage of each state was obtained under the six values of threshold S_b.

The results indicated that the percentage of the states showed a clear trend in the earlier generations (see Figure 6). Also, the results agreed with those obtained in the first set of simulations, i.e., the results showed that, below a given S_b threshold, state 0 disappeared and the SL survived S_b (values 4 to 7), but above a certain S_b threshold, states 1 and 2 disappeared and the SL consequently became extinct (values 8 and 9). An important finding was that in the cases where the SL became extinct, this extinction was reached quickly (in one or two generations) as the Gaelic-Arvanitika model predicts.

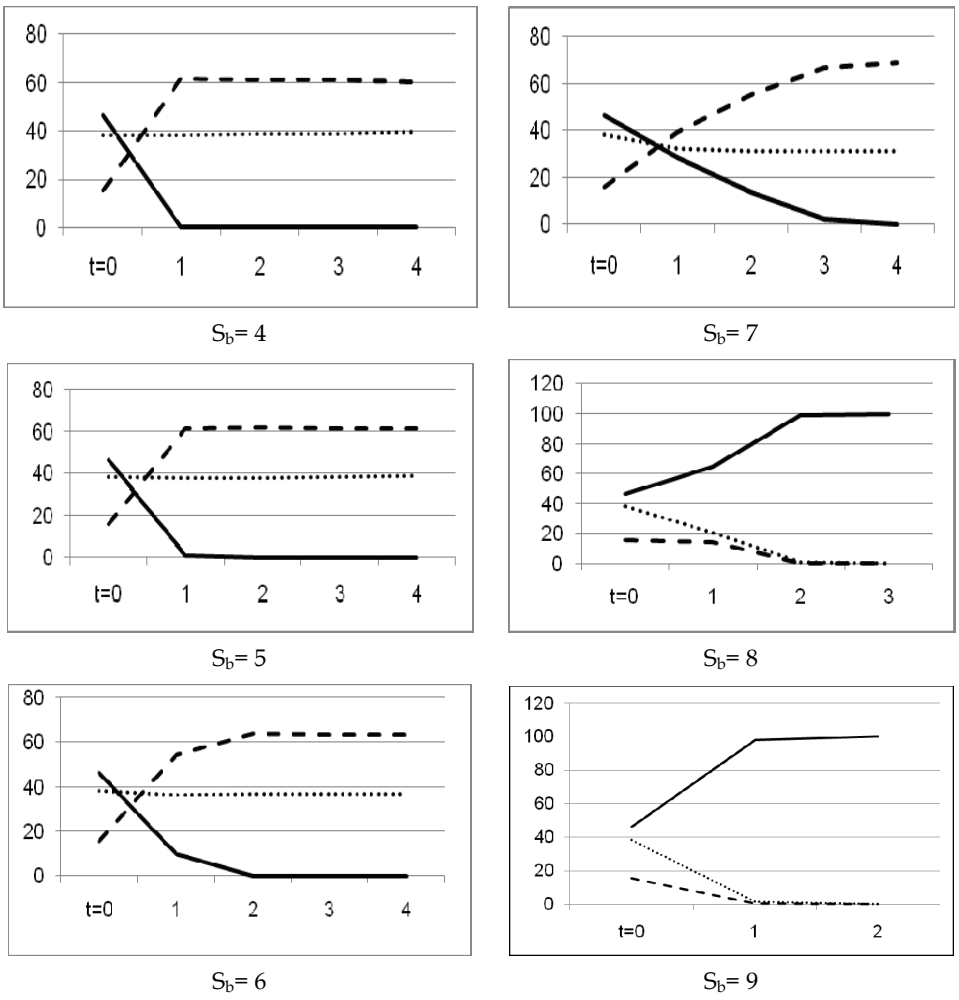


Fig. 6. Mean percentages of states 0 (solid), 1 (dashed) and 2 (dotted) for the values of threshold S_b when the automaton stabilized (values of S_b= 4 to 9) for the social context *home*. The x-axis indicates the generations of speakers. The percentage of initial values (*t=0*) is also shown. (Note that for the values S_b= 8 and S_b= 9, state 2 quickly reaches value 0).

The results of the two sets of simulations confirmed the importance of attitudes regarding the SL to determine individual speech behavior. If individuals are weakly engaged with the SL (according to our model a higher value of threshold S_b is required to use and eventually increase the use of the SL), the SL will disappear. However, if individuals are strongly engaged with the SL (according to our model a lower value of threshold S_b is required to use and eventually increase the use of the SL), the SL will survive. In this case, we also observed that the percentages of state 2 remained approximately constant, while the percentages of state 1 increased and those of state 0 decreased (Figures 5 and 6). Hence, the SL survived because the state-2 speakers continued speaking the SL, i.e., they did not become state-0 speakers, and the state-0 speakers used the SL because they became state-1 speakers, i.e., the monolinguals in the DL become bilinguals.

The results of the second set of simulations confirmed the results obtained in the first set, because values 4 to 7 of threshold S_b produced the extinction of state 0 (the monolinguals became state-1 bilinguals) and values 8 and 9 of threshold S_b produced the extinction of states 1 and 2 (all the bilinguals became monolinguals in the DL). That result remained constant across generations. Furthermore, the results of the second set of simulations confirmed a main prediction of the Gaelic-Arvanitika model, because the threatened language disappeared in few generations when transmission in the family failed (the extinction of states 1 and 2 occurred in only two generations).

4. Conclusion

As in our previous studies (Beltran et al., 2009, 2010), the results of the simulations using the empirical data of linguistic surveys showed the importance of the initial values of the speakers of the SL and their engagement with the SL (the percentage of initial states and value of the threshold S_b in our model) to the future of a SL when it coexisted with a DL. According our model, when high S_b values were set, states 1 and 2 completely disappeared, so the SL died out. However, when lower values were set, state 0 disappeared and the SL survived because all the individuals became bilinguals. The results also coincided in all social contexts tested by the simulations (at home, with friends, at traditional stores and at large shopping malls), because the values of the states were similar in the four contexts.

The results also agreed with the Gaelic-Arvanitika model. As stated above, a strong engagement of individuals with the SL produced the reversion of the language shift because all the monolinguals in the DL became bilinguals. Moreover, transmission of the SL to subsequent generations increased because the number of bilingual people who transmitted the SL grew. But weak engagement of individuals with the SL produced the extinction of the SL in only two generations, i.e., the results support the prediction of the Gaelic-Arvanitika model, which anticipated a quick language shift.

The results of our simulations provided some answers about the future of Catalan. Using the results of a language survey carried out in Catalonia on the effective use of Catalan to determine the initial size of the states (Idescat, 2008), the simulations confirmed that, given an initial size of the states, the value of threshold S_b (the engagement of individuals with the use of the SL) determined whether Catalan died out or not. Strong engagement of individuals (a low S_b value) with Catalan led many of the non-Catalan speakers to become bilingual (changing from state 0 to state 1). Thus, Catalan survived. Given the fact that an important issue related to language death is designing language policies to reverse language shifts, our results suggest that, together with government initiatives favoring the use of

Catalan, it will be necessary to implement language initiatives that favor speech behavior. Specifically, given the fact that the individual's engagement with the SL becomes critical in determining his or her speech behavior with regard to the SL, language initiatives should be aimed at convincing people to use Catalan even if there are few neighboring Catalan speakers. Moreover, these linguistic policies should be implemented as soon as possible, because the possible shift from the SL to the DL, from Catalan to Spanish in our example, is a rapid process.

As stated in the introduction, language extinction is a widespread social phenomenon that requires close attention from social scientists. Although future research should be improved in different ways (for example, the automaton should be applied to different examples of possible language shifts around the world), modeling the linguistic behavior of individuals by means of a cellular automaton has proven to be a useful tool for understanding language shift processes. Also, the study of language shifts based on a cellular automata approach can be a way to predict the future of threatened languages and, consequently, to design language policies to reverse the language shift process. Social simulation using cellular automata can therefore give us a new and promising framework for future theoretical and empirical development of language shift studies.

5. References

Beltran, F.S., Herrando, S., Ferreres, D., Estreder, V., Adell, M.-A. & Ruiz-Soler, M. (2009). Forecasting language shift based on cellular automata. *Journal of Artificial Societies and Social Simulation* [on line], 12, 3, 5. ISSN 1460-7425. Available http://jasss.soc.surrey.ac.uk/12/3/5.html.

Beltran, F.S., Herrando, S., Estreder, V., Ferreres, D., Adell, M.-A. & Ruiz-Soler, M. (2010). A language shift simulation based on cellular automata, In: *Handbook of Research on Culturally-Aware Information Technology: Perspectives and Models*, Blanchard, E.G. & Allard, D. (Eds.), 136-151, IGI Global, ISBN 978-1-61520-883-8, Hershey, PA.

Fishman J. A. (1991). *Reversing language shift. Theoretical and empirical foundations of assistance to threatened languages*, Multilingual Matters, ISBN 978-1-85359-121-1, Clevedon, UK.

Gilbert, N. (1996). Simulation as a research strategy, In: *Social science microsimulation*, Troitzsch, K. G.; Mueller U.; Gilbert, N. & Doran J. E. (Eds.), 448-454, Springer, ISBN 978-3540615729, Berlin.

Gilbert, N. & Toitzsch, K.G. (2005). *Simulation for the Social Scientis*, Open University Press, ISBN 0-335-21600-5, Berkshire, England (2nd edition).

Golspink, C. (2002). Methodological implications of complex systems approaches to sociality: Simulation as a foundation for knowledge. *Journal of Artificial Societies and Social Simulation* [on line], 5, 1, 3. ISSN 1460-7425. Avaliable http://jasss.soc.surrey.ac.uk/5/1/3.html

Hegselmann, R. (1996). Understanding Social Dynamics: The Cellular Automata Approach, In: *Social Science Simulation,* Troitzsch, K.G.; Muller, U.; Gilbert, N. & Doran, J.E. (Eds.), 282-306, Springer, ISBN 3-340-61572-5, Berlin.

Hegselmann, R. & Flache, A (1998) Understanding complex social dynamics: A plea for cellular automata based modelling. *Journal of Artificial Societies and Social Simulation* [on line], 1, 3, 1, ISSN 1460-7425. Available http://jasss.soc.surrey.ac.uk/1/3/1.html

Idescat (2008). *Enquesta d'usos lingüístics de la població* [Survey on Language Use], Secretaria de Política Lingüística, Generalitat de Catalunya, ISBN in process, Barcelona.

Latané, B. (1981). The psychology of social impact. *American Psychologist,* 36, 343-365, ISSN 0003-066X.

Latané, B. (1996). Dynamic social impact. In: *Modelling and Simulation in the Social Sciences from the Philosophy of Science Point of View*, Hegselmann, R.; Mueller, U. & Troitzsch, K.G. (Eds.), 285-308, Kluwer, ISBN 0-792-34125-2, Dordrecht, The Netherlands.

Latané, B., Liu, J.H., Nowak, A., Benevento, M. & Zheng, L. (1995). Distance Matters: Physical Space and Social Impact. *Personality and Social Psychology Bulletin,* 21, 8, 795-805, ISSN 0146-1672.

Latané, B., Nowak, A., & Liu, J.H. (1994). Measuring Emergent Social Phenomena: Dynamism, Polarization and Clustering as Order Parameters of Social Systems. *Behavioral Science,* 39, 1, 1-24, ISSN 1099-1743.

Latané, B. and Wolf, S. (1981). The Social Impact of Majorities and Minorities. *Psychological Review,* 88, 5, 438-453, ISSN 0033-295X.

Melià, J. L. (2004). Com es destrueix la llengua dels valencians: Un model binomial pels efectes de la regla de submissió lingüística [How the language of the Valencians is destroyed: A binomial model of the effects of the linguistic submission rule]. *Anuari de Psicologia de la Societat Valenciana de Psicologia*, 9, 1, 55-68. ISSN 1135-1268.

Mühlhäusler, P. (1996). *Linguistic Ecology: Language Change and Linguistic Imperialism in the Pacific Region*, Routlege, ISBN 0-415-05635-7, London.

Ninyoles, R. L. (2005). *Coneixement i ús social del valencià (síntesi de resultats)* [Knowledge and Social Usage of Valencian (Summary of Results)], Servici d'Investigació i Estudis Sociolingüístics, Direcció General de Política Lingüística de la Generalitat Valenciana, ISBN 84-482-3801-X,Valencia.

Nowak, A. & Lewenstein, M. (1996). Modeling Social Change with Cellular Automata. In: *Modelling and Simulation in the Social Sciences from the Philosophy of Science Point of View,* Hegselman, R.; Troizsch, K.G. & Muller, U. (Eds.), 249-285, Kluwer, ISBN 0-792-34125-2, Dordrecht, The Netherlands.

Nowak, A.; Szamrez, J. & Latané, B. (1990) From Private Attitude to Public Opinion: A Dynamic Theory of Social Impact. *Psychological Review,* 97, 3, 362-376, ISSN 0033-295X.

Sasse, H.-J. (1992). Theory of language death. In: *Language Death: Factual and Theoretical Explorations with Special Reference to East Africa*, Brezinger, M. (Ed.), Mouton de Gruyter, ISBN 3110134047, New York.

16

Cellular Automata Modelling of the Diffusion of Innovations

Gergely Kocsis and Ferenc Kun
Department of Theoretical Physics, University of Debrecen H-4010
Debrecen, P.O.Box: 5
Hungary

1. Introduction

Socio-economic and complex physical systems share several important features. Both are composed of a large number of interacting components where in most of the cases the precise form of the interaction is not known. In spite of this microscopic complexity, on the macro level such a state emerges which can be described in terms of a few parameters. Due to the collective behavior of the constituents of the system a universal macroscopic behavior emerges which does not depend anymore on the microscopic details of the system Helbing (2009). Technological development of socio-economic systems exhibits such universal aspects: irrespective of the field of economy, type of industry, technologies always evolve through cycles of birth, selection, disappearance and birth of the successor technology. The selection is made by the market which tests the capabilities of a technology and when it proves to be insufficient under the new circumstances, it is substituted by a newly born technology. The cyclic development of technologies gives rise to a logistic growth which can be described by only two parameters. Specific features of a given technology determine solely the value of the two parameters Rogers (1962).

During the last three decades efficient theories and models have been developed in statistical physics to describe the emergent behavior of complex systems. Methods have been worked out which can grasp the transition from microscopic complexity to the universal macroscopic behavior Helbing (2009); Sornette (2000). The theory of phase transitions, the renormalization group theory, the concept of self-organized criticality, dynamic critical phenomena and stochastic processes, and the theory of networks have been proven to be successful for complex system resulting in multitude of application in socio-dynamics as well Castellano et al. (2009); Gilbert (2008); Mahajan & Peterson (1985).

Cellular automata (CA) have been introduced in the field of socio-dynamics as an efficient approach for bottom-up models where individuals (agents) are the basic units of the system. Agents are described by a set of attributes, furthermore, they interact with each other and their social environment. For the diffusion of innovations the most important feature is that agents make decisions based on the influence they receive through word-of-mouth communications with their social partners and through some external information source (mass media). Recent investigations have shown that decision making in agent based models can be well described

by a set of rules and can be efficiently implemented in the framework of cellular automata Gilbert (2008); Kocsis & Kun (2008); Kun et al. (2007).
In this chapter we provide an overview of cellular automata modeling approaches to socio-economic systems with emphasis on the spreading of innovations. After summarizing the basic ingredients of CA we focus on the recent developments in the computer modeling of socio-economic systems. We outline the philosophy of bottom-up approaches of agent based models and describe typical set of CA rules which have been proven successful during the past years in the field. As a specific example, we present in details cellular automata for the spreading of those type of technological innovations whose usage requires so-called compatibility. These are for instance telecommunication technologies such as mobile phones, where a broad spectrum of mobile phone devices are offered by the market with widely different technological levels. Communication between two individuals, however, is the easiest when they use phones with nearly identical technological levels, since only in this situation they are able to benefit from capabilities such as MMS or Video messaging. We analyze the model analytically then set up CA rules of the model and present results of large scale computer simulations. The chapter is closed by an outlook summarizing possible future perspectives of the field.

2. Bottom-up approaches for the diffusion of innovations

Since Johann Louis von Neumann introduced it in order to study living biological systems in 1948 von Neumann (1948), cellular automata modeling has found a broad range of applications in the field of complex systems. The most widespread definition of cellular automata is that a CA is a finite number of finite state *cells* on a grid, which can change their *state* in discrete time steps according to the present state of their neighborhood. Usually the cells are placed on a square lattice with periodic boundary conditions such that each cell is affected only by its 4 (von Neumann neighborhood) or 8 (Moore neighborhood) neighbors. Classically the cells can hold two different states represented by 0 and 1. The update of cells' state is usually performed in a parallel way at the same time for each cell. The way how the state is changed defines the CA *rules*. Many eye-catching classical CA rules have became famous in the past, with more or less practical usage Wolfram (2002). Based on von Neumann's basic mathematical concepts, CA models became the basis of the so called simulation games in the 1970s. The most famous example of such games is John Horton Conway's "Game of Life" Gardner (1970). In spite of the successful applications of CA in these games, they gained popularity only in the 1980s through the work of Stephen Wolfram, who gave an extensive classification of CA as mathematical models for self-organizing statistical systems Wolfram (2002). Wolfram applied cellular automata to a huge number of scientific areas e.g. biology, physics, sociology, etc.
The use of cellular automata in the field of diffusion phenomena tracks back to these times as well Grassberger (1984), however, the effective power of CA in modeling diffusion could only be revealed after the revolutionary growth of computing power in the 1990s. By the end of the century CA simulation of diffusion models became an elementary tool in the field, and till today, in most of the cases CA based simulations represent the basic numerical methods in order to validate the analytical predictions of diffusion models. In order to observe the headway of CA modeling in a more specific field, one can take the case of *diffusion of innovations*, which has a history going back to the 1960s, but has an ever increasing popularity nowadays as well Guardiola et al. (2002); Helbing et al. (2005); Llas et al. (2003). The first

edition of Everett M. Rogers' pioneering book in 1962 used to be called as the starting point of innovation diffusion related research Rogers (1962). Currently the book is at its fifth edition updated and extended with up to date results and case studies. Besides Rogers' book one can get an interesting insight into the past and present of innovation diffusion from numerous recapitulatory papers of Mahajan Mahajan & Peterson (1985) or from the work of Castellano et. al. Castellano et al. (2009).

In his book, Rogers defines diffusion of an innovation as the process by which that innovation "is communicated through certain channels over time among the members of a social system". As a definition of innovation it says "innovation is an idea, practice, or object that is perceived as new by an individual or other unit of adoption" Rogers (1962). These definitions show that innovation diffusion gathers all the processes where something new spreads over a social system.

Cellular automata have successfully been applied to investigate the diffusion of innovations in socio-economic systems. CA approaches in socio-dynamics reflect the *bottom-up* modeling philosophy, *i.e.* agents are introduced which represent individuals of the society Gilbert (2008). Agents have to be characterized by a well-defined finite set of variables which in principle should be measurable in sociometric sense. The variables are defined such that they describe up to some extent the rational and irrational (emotional) aspects of agents' behavior from the viewpoint of the scope of the model (for instance, opinion formation before political election or spread of technologies on the market after new inventions are introduced).

Such agent-based models are definitely disordered in the sense that the variables describing agents must have broad variations in the system. The distribution of agents' properties should again reflect some general tendencies in the society based on sociological surveys.

The interaction of agents is rather complex, certainly much more complicated than the interaction of particles in any physical systems. In general, it is very difficult, therefore, to cast the interaction law in a closed mathematical form. For the sake of simplicity, two limiting cases can be formulated: (i) absolutely rational agent where the interaction means taking a well-defined decision based on the surroundings. Such an interaction-decision rule implies a deterministic time evolution of extended sociodynamic systems starting from an initially disordered state. (ii) absolutely irrational agent whose decision is perfectly random, the interacting partners can only affect the degree of randomness of the change of agents' variables compared to the preceding state. Bounded rationality is a decision mechanism which lies between the two extreme cases discussed above. Obviously, this is much more realistic but addresses serious mathematical problems to represent a decision mechanism which captures both deterministic (rational) and probabilistic (irrational or emotional) aspects.

Time evolution of the system is obtained by prescribing an appropriate dynamics of the system. The "dynamics" can be formulated in terms of decision rules according to which agents can change their state as time elapses. An important point of such agent-based model constructions of sociodynamics is if the dynamic rule is deterministic where disorder enters only through the disordered initial state of agents' properties. Such deterministic dynamics can be formulated in terms of cellular automata. The other limiting case is the stochastic dynamics similar to the dynamics of finite temperature systems in physics. Such dynamics can be implemented in the form of Monte Carlo simulations such as importance sampling with the Metropolis algorithm Gilbert (2008).

In this Chapter we present a study of the spreading of innovations in socio-economic systems using a bottom-up approach as described above which is implemented in a cellular automata framework. We focus on those technologies where the practical value, the usability

or advantages of the technology for the user depends on the number of social partners already using the technology. Telecommunication technologies are of that kind, since a mobile phone is rather useless if there is nobody to call with. The CA rules of the model are based on the cost minimization requirements, i.e. agents can change their technological level if it provides reduction of the communication costs. As a key ingredient we assume that the mechanism of spreading is the copying, i.e. agents purchase products of technologies copying the product of one of their interacting partners. After presenting the details of the model construction we analytically investigate simplified cases then we present results of realistic cellular automata simulations for both regular lattices and complex network topologies of social contacts.

3. Model for the spreading of technologies in socio-economic systems

Technological evolution of socio-economic systems is composed of two phenomena Mahajan & Peterson (1985); Rogers (1962); Weidlich (2000): *(i)* New products, ideas, working methods, emerge as a result of *innovations* which are then used by the market. *(ii)* Successful technologies *spread* in the system resulting in an overall technological progress. One of the key components of this spreading of successful technologies is the copying, i.e. members of the system adopt technologies used by other individuals according to certain decision mechanisms. Decision making is usually based on a cost-benefit balance so that a technology gets adopted by a large number of individuals if the upgrading provides enough benefits Rogers (1962).
In the present work we focus on the spreading process assuming that several technologies coexist in the system providing alternative solutions for the same practical problem. We introduce a simple agent-based model of the spreading process restricting the investigations to technologies where "networking" plays a crucial role Rogers (1962), i.e. the technology is used for communication/interaction where a certain compatibility is required. In real life it can be often seen that in some cases less advanced technologies rule the market and they still proliferate even if a new, somehow better technology appears. On the other hand if a large enough number of users start to use a new technology sooner or later the whole community follows them making the older technology disappear. The focus of our study is on finding answers to the questions of what specific criteria have to be fulfilled in order to make a technology successful in the market.

3.1 Cost of communication

In our model we represent the socio-economic system by a set of agents which posses products that may be of different technological levels and use it to cooperate with each other. The technological level of the products (e.g. a mobile phone or any device which can be used for communicating with others) is described by a real variable τ in such a way that more advanced technologies are characterized by a higher value of τ. The technology held by an agent is used for communication/interaction between its social contacts. It is easy to understand that communication is the easiest if the interacting partners have devices with nearly the same technological level. The usage of devices of highly different technological levels may cause difficulties in the interaction which result in additional costs. As a more specific example let us consider the case of SMS communication. Old mobile phones of lower technological level could send SMS messages only with a maximal length of 160 characters, however, for the new ones the allowed length is three times larger. Sending a long message between an old phone and a new one is possible, of course, because we only have to split our text into three parts, but naturally this procedure makes communication much more

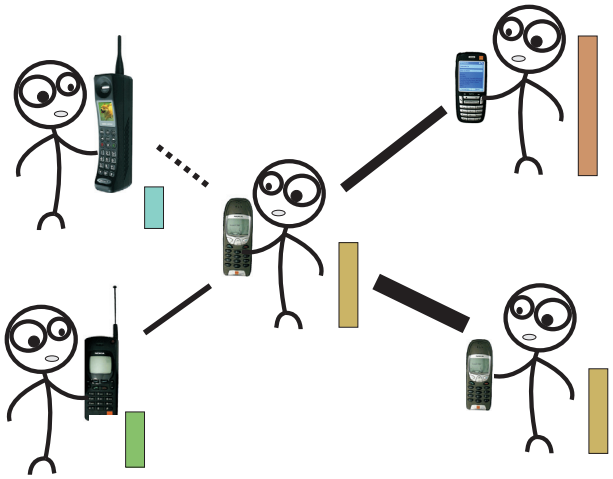


Fig. 1. Demonstration of the basic ideas of the model construction. Agents use different level technologies (mobile phones) to communicate with each other. The different capabilities of the devices (SMS, MMS, video-phone) induce difficulties, i.e. communication is the easiest between devices of the same technological level (this is indicated by the black lines between agents). The height of the colored rectangle indicates the "technological level" of the device of the corresponding agent.

difficult and uncomfortable. In the opposite direction we have to notice that our message will be automatically split up into pieces. Agents using mobile phones are presented in Fig. 1. This simple example clearly illustrates that the source of difficulties is the difference in the technological levels of the devices used for communication and they would not occur if the two partners would use equally advanced technologies. It has to be emphasized that in our modeling approach *cost* does not only mean the money one has to pay for the services or for the device, but it covers all types of difficulties (including also financial ones) that can affect the quality of the communication (e.g. time, convenience etc.) Kocsis & Kun (2008); Kun et al. (2007).

Based on the above arguments it is reasonable to assume that the cost C induced by the communication of agents i and j is a monotonic function of the difference of the technological levels $|\tau_i - \tau_j|$. For the purpose of the explicit mathematical analysis we consider the most simple functional form and cast the cost of cooperation into the following form Kun et al. (2007)

$$C(i \to j) = a|\tau_i - \tau_j|. \tag{1}$$

Equation (1) expresses that having products of different technological levels (having different values of τ) incurs cost, while interaction with devices of the same technological level is cost free. This crude assumption describes a socio-economic system which favors the local communities being at the same technological level. Producers fabricate and introduce new communication devices on the market with the goal to provide solutions of possible problems, difficulties customers may have. This generic tendency of technological development can be captured in the model by setting appropriate values for the multiplication factor a of the cost

function Eq. (1). Hence, we assume that the value of a depends on the relative technological level of interacting agents as

$$a = \begin{cases} a_1, \text{ if } \tau_i > \tau_j \\ a_2, \text{ if } \tau_i < \tau_j \end{cases} \qquad where \qquad a_1 < a_2 \tag{2}$$

which clearly favors the higher technological level of users. As a result of the condition $a_1 < a_2$ using a more advanced technology than the surroundings $\tau_i > \tau_j$ implies lower costs compared to the opposite case. Note that as a result of this condition the cost function is not symmetric with respect to agents i and j. This property of C is expressed by the arrow $\rightarrow$ in the argument so that $C(i \rightarrow j)$ defines the cost of agent i arising due to the cooperation with agent j which is not equal to the cost of agent j, i.e. $C(i \rightarrow j) \neq C(j \rightarrow i)$. Knowing the cost of interaction between communication partners we can now define the total cost of a given agent in the model system. If agent i has n collaborating partners with technological levels $\tau_1, \tau_2, \ldots, \tau_n$, the total cost of its collaboration can be obtained by summing up the cost function Eq. (1) over all connections

$$C(i) = \sum_{j=1}^{n} C(i \rightarrow j). \tag{3}$$

3.2 Time evolution

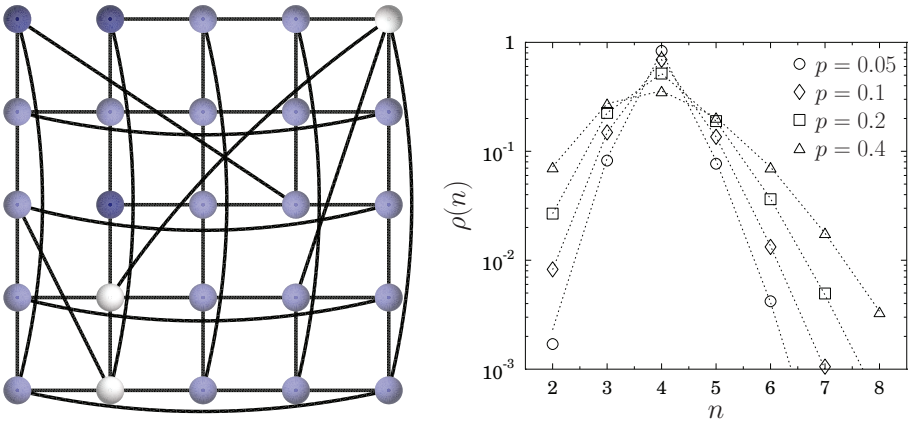
In order to reduce their costs, agents are assumed to be able to change their technological level which results in a non-trivial time evolution of the system. Our approach focuses on the spreading of technologies so that agents do not invent new products, the possible level of technologies are determined by the initialization of agents' characteristics. The driving force of evolution in the system is the tendency that agents try to optimize the cost of their communication reducing the value of C. To achieve this goal, however, they can only adopt/copy technologies choosing the one of their interacting partners

$$C^{t+1}(i) = \min\{C(\tau \in \{\tau_i^t, \tau_1^t, \tau_2^t, \ldots, \tau_n^t\}\}, \tag{4}$$

where the copy is always executed if it provides cost reduction $C^{t+1}(i) < C^t(i)$.
It is assumed in the model, that adopting a technology does not induce costs, i.e. no money is required to buy the new products, thus agents can change their technological level anytime if the change provides cost reduction in the future. The financial status of agents, the amount of money, statistics of income in the society do not play any role in the present setup of the model system but of course it can easily be implemented. This rejection-adoption process based on local cost minimization results in the spreading of the adopted technologies while the rejected ones disappear from the system. Our model emphasizes that the key component of the spreading of innovations is the copying with the aim of ensuring compatibility and hence, reduction of the difficulties (cost of communication). Note that in the model there is no intrinsic advantage of using more advanced technologies, the cost is the same independently of the technological level when consensus has been reached Kocsis & Kun (2008); Kun et al. (2007).

3.3 Topology of social contacts

For agent based socio-dynamics models it is a crucial point to implement a realistic representation of agents' social contacts. We apply the Watts-Strogatz rewiring algorithm Watts & Strogatz (1998) to generate complex networks in order to mimic the topological features of the social contacts in real social systems Kocsis & Kun (2008). We start the process with a square lattice of agents with periodic boundary conditions. The rewiring process is performed such that we take every edge on the lattice and remove it with a probability p. After that the connection is re-established between two agents selected randomly with a uniform distribution.



(a) (b)

Fig. 2. (a) Complex networks of agents with long range connections are obtained by rewiring a square lattice. The color of the nodes corresponds to the number of their connections (lighter blue stands for a higher degree). (b) Degree distribution $\rho(n)$ of rewired square lattices for different rewiring probabilities p. Increasing the value of p the distribution gets broders but the position of the maximum does not change.

The rewiring procedure is illustrated by Fig. $2(a)$. The rewiring process has the consequence that degrees different from 4 occur in the social network with a certain probability and the topology of the system is changed from short ranged ($p = 0$) to random graphs ($p \to 1$) with long range connections Watts & Strogatz (1998). The distribution ρ of the degree of agents n, i.e. the number of connections of the agents, can be determined analytically as the convolution of a binomial and a Poissonian distribution Albert & Barabasi (2002)

$$\rho(n) = \sum_{s=0}^{min(n-k,k)} \binom{k}{s} (1-p)^s p^{k-s} \frac{(pk)^{n-k-s}}{(n-k-s)!} e^{-pk}, \tag{5}$$

where k denotes the half of the average degree and n is the observed degree. The weights of the binomial and Poissonian components are in a linear relation to the rewiring probability p. Fig. $2(b)$ illustrates degree distributions of rewired square lattices, generated with the Watts-Strogatz method using various values of the rewiring probability p. It can be observed

that with the increase of p, however the average degree remains constant $\langle n \rangle = 4$, the distribution $\rho(n)$ gets broader, increasing the polydispersity of social contacts. The Watts-Strogatz type complex networks have been proven to be very useful in studying social phenomena Watts (1999). For the spreading of technological development the main limitation we face is that the network has a static structure, it does not evolve during the diffusion process. However, this frozen structure allows us to make an efficient cellular automata implementation of the system.

4. Analytical investigation of the model

In order to understand the decision making mechanism, how agents select the technology to adopt, and to reveal which features of the system determine the success of technologies on the market, it is useful to study simplified configurations by analytical calculations. First we analyze the ideal case when all agents communicate with each other irrespective of their spatial distance then we consider isolated local communities of relatively small size. We work out a master equation approach which reveals interesting fixed points in the parameter space of the system governing the long term time evolution of technological levels and the overall technological progress of the system.

4.1 Mean field versus local interaction

As a starting scenario let us assume that the system is composed of a large number of agents which have randomly distributed technological levels in an interval $\tau_{min} \leq \tau \leq \tau_{max}$ with a probability density $p(\tau)$ and distribution function $P(\tau) = \int_{\tau_{min}}^{\tau} p(\tau')d\tau'$. If we assume infinite range of interactions, all agents are connected with each other so the cost of interaction of an agent of technological level τ can be cast into the form

$$C(\tau) = a_1 \int_{\tau_{min}}^{\tau} (\tau - \tau')p(\tau')d\tau' + a_2 \int_{\tau}^{\tau_{max}} (\tau - \tau')p(\tau')d\tau' \tag{6}$$

as a function of τ. In the next time step the agent will change its technological level from τ to that τ^*, which minimizes the cost function Eq. (6), i.e. $dC/d\tau|_{\tau^*} = 0$. The technology optimizing the cost can finally be obtained as the solution of the equation

$$P(\tau^*) = \frac{1}{1 + 1/r}, \tag{7}$$

where $r = a_2/a_1$ is the ratio of the two cost factors a_1 and a_2 and P is the cumulative probability distribution of the technological levels of agents in the initial configuration. A very interesting outcome of the above calculations is that the result does not depend on the precise value of the cost factors a_1 and a_2 but only on the the relative amount of advantages $r = a_2/a_1$ more advanced technologies provide with respect to the lower level ones. Since the range of interaction is infinite, all agents make the same decision, thus after a single time step all agents adopt the same technology τ^* leading to the end of the evolution of the system. In the special case of $r = 1$ (when a higher technological level does not provide any advantages), the system adopts the median $\tau^* = m$ of the initial distribution of technologies $p(\tau)$ Sornette (2000). It is interesting to note that the optimal choice $\tau^*(r)$ is a monotonically increasing function of r, however, the most advanced technology τ_{max} is solely chosen in the limiting case $lim_{r\to\infty}\tau^*(r) = \tau_{max}$. At any finite value of $r > 1$ the large number of agents of low level technologies can force the system to stay at a lower technological level which shows that for

the overall technological progress of the system strongly connected social networks may be disadvantageous.

As a next step let us focus on what happens in a more complex society at the level of small sized local communities. We consider a finite community of n agents with technological levels $\tau_1 < \tau_2 < \cdots < \tau_n$ communicating with each other. The cost of collaboration between agent i of technological level τ_i with the other $n-1$ agents reads as

$$C(\tau_i) = a_1 \sum_{j=1}^{i-1} (\tau_i - \tau_j) + a_2 \sum_{j=i+1}^{n} (\tau_j - \tau_i). \tag{8}$$

In the next time step the agent decides to adopt that technology which minimizes the cost function Eq. (8) among the $n-1$ possibilities. Analytical calculations show again that this decision is solely determined by the value of r, namely, the ith highest technological level is adopted $\tau^* = \tau_i$ when r falls in the interval

$$\frac{i-1}{n-i+1} < r < \frac{i}{n-i} \qquad for \qquad 1 \le i < n, \tag{9}$$

$$n-1 < r \qquad for \qquad i = n. \tag{10}$$

It can be seen from the above equations that the limits of the sub-intervals of r to choose the ith and $n-(i-1)$th largest τ are symmetric with respect to $r = 1$. The most advanced technology $\tau^* = \tau_n$ of the available ones is adopted only if r exceeds the number of interacting partners $r > n-1$. Of course, the actual value of τ^* is not determined by the above equations, so that in a system composed of a large number of local communities of agents with randomly distributed τ values a complex time evolution emerges, which is locally governed by the equations Eq. (9) and Eq. (10).

4.2 Master equation approach

As the next step of complexity let us examine the case where only two products are present in the system with different technological levels $\tau_0 < \tau_1$ but social contacts have a realistic topology characterized by the degree distribution function ρ Eq. (5). For simplicity we set the technological levels to $\tau_0 = 0$ and $\tau_1 = 1$ without loss of generality. At the beginning $t = 0$ let the fraction of agents having products of the two technological levels be $\phi^0_{t=0}$ and $\phi^1_{t=0} = 1 - \phi^0_{t=0}$, respectively. Our goal is to work out a master equation approach to determine the long term time evolution of the system varying the ratio of cost factors r in a broad range and the topology of social contacts controlled by the value of the rewiring probability p. Since only two technological levels are present in the system comprehensive description can be given by determining the time dependence of the fraction of agents ϕ^0_t and ϕ^1_t.

We assume that members of local communities are randomly scattered all over the system with a perfect mixing. The assumption implies that any clusterization of agents according to their technological level is omitted in the approach so that no spatial correlations can arise at this level of description. When an agent of technological level τ_0 communicates with its social partners, from the cost function Eq. (3) presented in Section 3, we can derive the minimum number k of neighbors having technological level τ_1 that make the agent switch to the other technological level τ_1

$$k > n \frac{1}{1+r} \equiv n p_h, \tag{11}$$

where p_h denotes the minimal fraction of neighbors holding technologies of $\tau_1 = 1$ necessary to induce the transition. Based on Eq. (11) we can determine the transition probability $p_t^{0\to1}$ that an agent, with technological level τ_0 changes to τ_1 at time t

$$p_t^{0\to1} = \sum_n^{\infty} \rho_n \sum_{k=\lceil np_h \rceil}^{n} \binom{n}{k} (1-\phi_{t=0}^0)^k (\phi_{t=0}^0)^{n-k}. \tag{12}$$

Here ρ_n denotes the degree distribution of the underlying topology and $\lceil . \rceil$ represents the ceiling function, i.e. the nearest integer being greater or equal. In the above equation Eq. (12) we used the assumption that in every time step the system starts from a totally random state and just the values of ϕ_t^0 and ϕ_t^1 are changing over time.
In the opposite case we wish to derive the probability that a given agent will change its technological level from $\tau_1 = 1$ to $\tau_0 = 0$. The analytical form in this case barely differs from the previous one due to the symmetric nature of the cost factor r presented in Eq. (2). In order to obtain this probability we only have to change the limits of the second sum of Eq. (12)

$$p_t^{1\to0} = \sum_n^{\infty} \rho_n \sum_{0}^{k=\lceil np_h \rceil - 1} \binom{n}{k} (1-\phi_{t=0}^0)^k (\phi_{t=0}^0)^{n-k}. \tag{13}$$

Note that in this special case of the model the equality $p_t^{1\to0} = 1 - p_t^{0\to1}$ holds since there are only two technological levels presents in the system. Knowing the fraction of technological levels ϕ_t^0 and ϕ_t^1 and the transition probabilities $p_t^{0\to1}$ and $p_t^{1\to0}$ at time t we can obtain discrete evolution equations for the fractions of technological levels

$$\phi_{t+1}^0 = \phi_t^0 + p_t^{1\to0}(1-\phi_t^0) - p_t^{0\to1}\phi_t^0, \tag{14}$$

$$\phi_{t+1}^1 = 1 - \phi_{t+1}^0 = \phi_t^1 + p_t^{0\to1}(1-\phi_t^1) - p_t^{1\to0}\phi_t^1. \tag{15}$$

After specifying the initial fractions $\phi_{t=0}^0$ and $\phi_{t=0}^1$ and the cost parameter r we can follow the evolution of the system by iterating Eqs. (14,15). The above master equations have also the advantage that the most important features of the time evolution can be extracted by analytic means.

4.2.1 Stable and instable fixed points

In order to reveal the long term time evolution of the system it is crucial to investigate the fixed points of the master equations Eqs. (14,15). Fixed points of the evolving fractions of technological levels τ are defined by the condition $\phi_{t+1}^\tau = \phi_t^\tau$, i.e. the fixed point is reached when a frozen state is attained. Here the technological level τ can be 0 or 1. Our goal is to identify well defined regimes of initial fractions $\phi_{t=0}^0$ and $\phi_{t=0}^1$ for different values of the cost factor r starting from which the system can converge to different final states. For simplicity, let us consider first Eq. (15) and analise the case when all agents have $n = 4$ social contacts. Applying the fixed point condition at $t = 0$, it can easily be seen that the iteration equation Eq. (15) has either two or three fixed points depending on the value of r. For the parameter ranges $r < 1/3$ and $r > 3$ there are two fixed points, a stable and unstable one which characterize homogeneous final states of the system where only one of the technologies survives

$$\phi_{c1}^1 = 0, \quad \text{and} \quad \phi_{c2}^1 = 1. \tag{16}$$

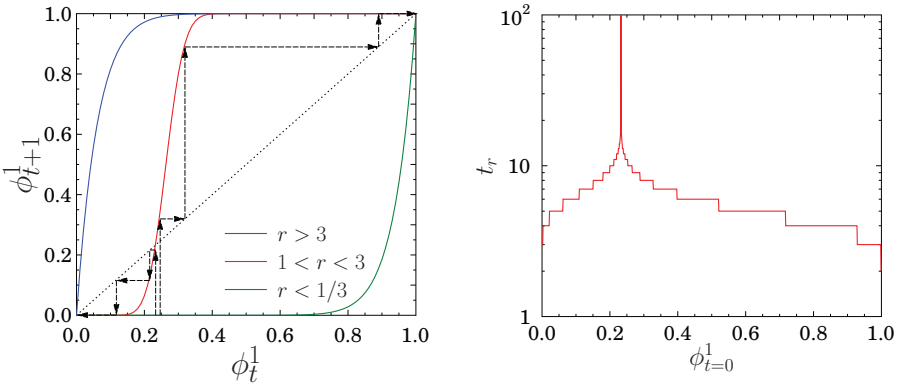


Fig. 3. (a) The function $\phi^1_{t+1}(\phi^1_t)$ for a square lattice of agents where the number of interacting partners is fixed $n = 4$ for all agents. Three curves are presented which are typical for the corresponding ranges of the cost factor r. For the red curve obtained for $r = 1.4$ iterations of Eq. (15) are presented starting from two different initial conditions $\phi^1_0 = 0.23$ and $\phi^1_0 = 0.25$. Following the dashed lines it can be observed that the system evolves into homogeneous final states. Note that the closer we start to the fix point, the longer it takes for the system to reach the final homogeneous state represented by the two stable fixed points. (b) The relaxation time t_r as a function of the initial fraction $\phi^1_{t=0}$. Approaching the unstable fixed point $\phi^1_{c3} = 0.232$ the relaxation time t_r diverges.

Analytical and numerical calculations showed that for $r < 1/3$ the first fixed point is stable while the other one is unstable. The result implies that the success of the lower level technology is guaranteed in the system since the range of attraction of the stable fixed point $\phi^1_{c1} = 0$ is the interval $\phi^1_0 \in [0, 1)$. In this range of the cost factor r the system always converges to the homogeneous state $\phi^1_t = 0$, i.e. the higher level technology disappears from the system irrespective of the initial fractions except for the case $\phi^1_0 = 1$. For $r > 3$ the fixed points are the same given by Eq. (16), however, their stability features change, namely, $\phi^1_{c1} = 1$ is stable with the range of attraction $\phi^1_0 \in (0, 1]$. This implies again the convergence to a homogeneous final state where now the lower level technology completely disappears and the system experiences technological progress.

It is very interesting to note that for the parameter ranges $1/3 < r < 1$ and $1 < r < 3$ the system has three fixed points: two fixed points characterize the homogeneous final states given by Eq. (16) as discussed above, however, they are both stable in these ranges of r. The third fixed point ϕ^1_{c3} can be determined from the iteration equation by considering

$$\phi^1_{t=0} = \phi^1_{t=0} + p^{0\to1}_0(1 - \phi^1_{t=0}) - p^{1\to0}_0\phi^1_{t=0}. \tag{17}$$

Rearranging Eq. (17) leads to

$$p^{0\to1}_0(1 - \phi^1_{t=0}) = p^{1\to0}_0\phi^1_{t=0} \tag{18}$$

and since $p_0^{0\to1} = 1 - p_0^{1\to0}$ holds, it follows that

$$p_0^{0\to1}(1 - \phi_{t=0}^1) = (1 - p_0^{0\to1})\phi_{t=0}^1, \tag{19}$$

which then yields

$$p_0^{0\to1} = \phi_{t=0}^1. \tag{20}$$

The final expression Eq. (20) implies that the fractions of the two technological levels will keep unchanged during the whole time evolution of the system if the probability that an agent with technological level $\tau_0 = 0$ changes to technological level $\tau_1 = 1$ equals to the initial fraction of agents of technological level $\tau_1 = 1$. The numerical solution of Eq. (20) gives the third fixed point $\phi_{c3}^1 = 0.768$ and $\phi_{c3}^1 = 0.232$ for $1/3 < r < 1$ and $1 < r < 3$, respectively. The third fixed point proved to be unstable, i.e. the state characterized by ϕ_{c3}^1 is only attained by the system if the initial condition is set as $\phi_{t=0}^1 = \phi_{c3}^1$. From any other initial state the system converges to one of the two stable fixed points where only one of the technologies survives.

The iterations of the dynamic equation Eq. (15) are illustrated in Fig. 3(a) for two different initial values of $\phi_{t=0}^1$ with the cost parameter $r = 1.4$ assuming that all agents have 4 interacting partners. The third fixed point ϕ_{c3}^1 can be identified as the intersection of the curve of $\phi_{t+1}^1(\phi_t^1)$ and of the straight line with slope 1. It can be observed that iterations starting on the left side of the fixed point $\phi_0^1 < \phi_{c3}^1$ always converge to the final state of $\phi_t^1 = 0$ leading to disappearance of the corresponding technology. However, iterations starting at a high enough initial fraction $\phi_0^1 > \phi_{c3}^1$ lead to the final dominance of the higher level technology $\phi_t^1 = 1$. The results demonstrate that the two fixed points $\phi_{c1}^1 = 0$ and $\phi_{c2}^1 = 1$ are stable and their ranges of attraction are the intervals $[0, \phi_{c3}^1)$ and $(\phi_{c3}^1, 1]$, respectively.

A very important outcome of the master equation analysis is that starting from a random configuration of two competing technologies the dynamics of the system leads to a homogeneous final state where only one of the technologies survives. However, depending on the value of the starting fractions of technologies the evolution process can even take a very long time. Figure 3(b) presents the relaxation time t_r, i.e. the time needed to reach the homogeneous final state as a function of the initial fraction $\phi_{t=0}^1$. It can be observed that approaching the third fixed point the relaxation time t_r diverges $t_r \to \infty$, which demonstrates that the inhomogeneous state of competing technologies can have a very long lifetime. This feature of the model describes the natural phenomenon that the competition of products on the market is much more tense and takes longer if the presence of products is balanced. The initial fraction can be controlled by the advertising activities of the producers.

It is a very interesting question to investigate how the complex topology of social contacts affects the spreading of technologies. Complex networks of agents generated by the Watts-Strogats method can be captured in the framework of the master equation approach by inserting the degree distribution Eq. (5) into the generic expression of the transition probabilities Eq. (12). In Figure 4 we present the value of the unstable fixed point ϕ_{c3}^1 in the range of the cost factor $1 < r < 3$ varying the rewiring probability p of the network in the interval $[0, 1]$. The rewiring method was applied to a square lattice so that in the figure $p = 0$ represents the regular square lattice, while in the limit $p \to 1$ a random graph is obtained. It can be observed that for $p = 0$ the value of the fixed point does not depend on r in the range considered (see also the above analysis). However, as the rewiring probability increases, i.e. as the degree distribution gets broader, two important changes appear: ϕ_{c3}^1 has a continuous dependence on the value of p, and the interval of r splits up into several sub-intervals inside

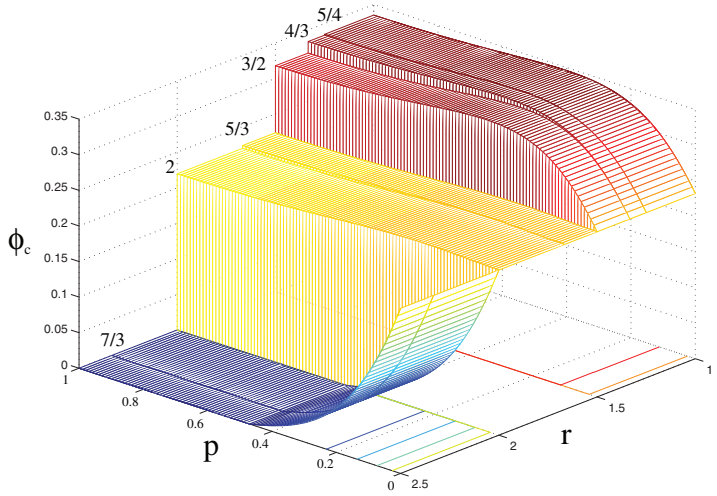


Fig. 4. The unstable fixed point of the system ϕ^1_{c3} as a function of the rewiring probability p in the range of the cost factor $1 < r < 3$. The polidispersity of the number of connections makes the system more sensitive to advantages provided by more advanced technologies. The numbers on the surface plot provide the interval limits of r inside which ϕ^1_{c3} is constant.

which ϕ^1_{c3} has different values. This simple master equation analysis demonstrates that the topology of social contacts may have a strong effect on the spreading process of technologies in social systems. More details on the effect of the underlying social network will be revealed by cellular automata simulations in the next section.

5. Cellular automata simulations

To be able to carry out analytical investigations of the model system in the previous section, serious simplifications had to be applied. The advantage of the approach is that interesting characteristic quantities could be obtained in closed analytic forms, important global features could be revealed, however, the results are limited either by the range of interaction, simplified topology of social contacts, bimodal distribution of initial technological levels, or by neglecting any spatial correlation (clusterization) of agents according to their technological levels. In order to analyze the time evolution and spatial structure of the model system in its entire complexity, we perform computer simulations using cellular automata techniques. As a first case we consider a set of agents organized on a square lattice of size $L \times L$ with nearest neighbor interactions. Initially agents have randomly distributed technological levels between 0 and 1 with uniform distribution

$$p_0(\tau) = 1, \qquad and \qquad P_0(\tau) = \tau, \qquad for \quad 0 \leq \tau \leq 1. \tag{21}$$

In our simulations we assume periodic boundary conditions, thus all agents of the lattice have four interacting partners. The rejection-adaptation dynamics based on the cost minimization

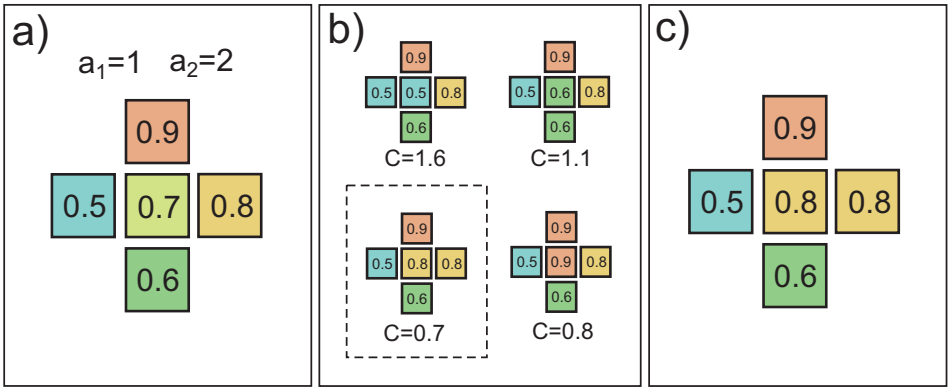


Fig. 5. Illustration of the update rule of cellular automata. (a) presents the current configuration of an agent with its 4 interacting partners. The value of the technological level τ and of the two cost factors a_1 and a_2 are given. (b) demonstrates that the agent in the middle has 4 possibilities to copy the technology of one of its social partners. The agent will choose the one which provides the lowest cost $C = 0.7$. (c) The final configuration after decision making. The color of the squares represents the technological level which also corresponds to the color code of Fig. 1.

results in a non-trivial time evolution of the system which is followed by computer simulations treating the system locally as a cellular automaton. In the simulations parallel update is used, i.e. all agents try to minimize their costs in each time step assuming that their interacting partners keep fixed. This parallel dynamics is one of the sources of the complex behavior of the system.

If at time t the technological level of agent i which has n neighbors with technological levels $\tau_1, \tau_2, \ldots, \tau_n$ is τ, the CA rule to get its technological level in time $t+1$ reads as

$$\tau_i^{t+1} = \tau_j, \tag{22}$$

where j denotes the neighboring agent whose technological level is the most worthy to copy for agent i, i.e.

$$C'(i,j) = min\{C'(i,1), C'(i,2), \ldots, C'(i,n)\}, \tag{23}$$

where $C'(i,j)$ is the cost of agent i assuming that its technological level has been replaced with the technological level of neighbor j. The update rule of our cellular automata is illustrated in Fig. 5 on a square lattice. Snapshots of the cellular automata time evolution of the evolving system are presented in Fig. 6 for a square lattice of size $L = 100$.

Applying the analytical results of Eq. (9) and Eq. (10) for the specific case of $n = 4$, the agents will always copy the first, second, third or fourth highest τ of their local interacting partners when the value of the parameter r falls in the intervals $0 < r < 1/3$, $1/3 < r < 1$, $1 < r < 3$, $3 < r$, respectively. (Note that the behavior of the system is symmetric with respect to $r = 1$.) Since the dynamics of the system governed by the cost minimization mechanism favors local communities to have products of the same technological level, the agents tend to form clusters with equal τ at any value of r. A very interesting special case is $r = 1.0$, when being more advanced than the surroundings does not provide any advantages, it can be seen in

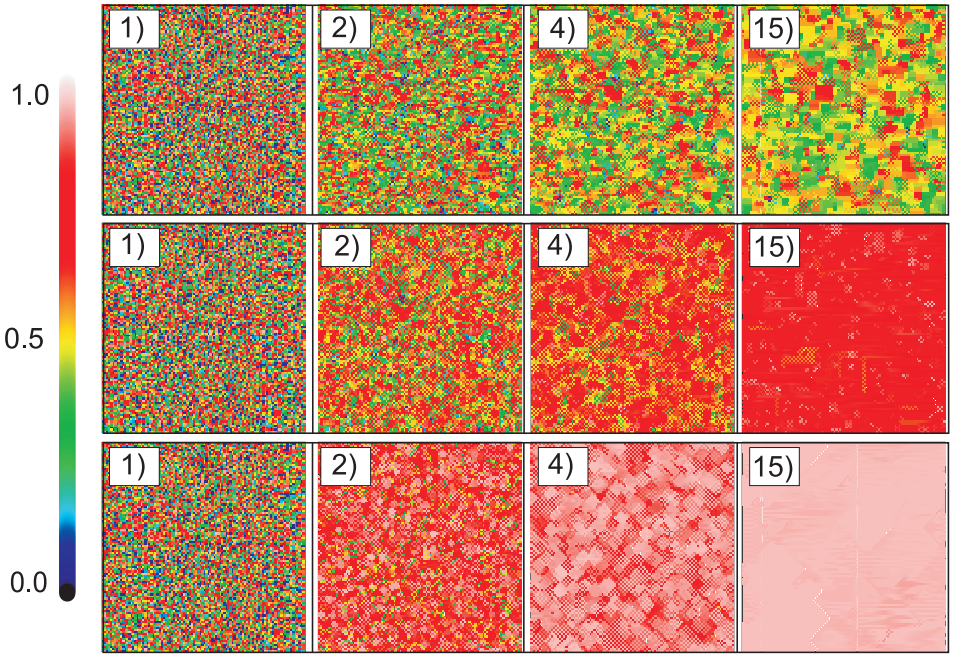


Fig. 6. Snapshots of CA simulations of the model on a square lattice of size $L = 100$ for different values of the cost factor: $r = 1$ *(top)*, $r = 2$ *(middle)*, and $r = 4$ *(bottom)*. The color code represents the actual technological level of the agents. The numbers indicate the time step in which the snapshot was taken.

Fig. 6*(top row)* that the system evolves into a frozen cluster structure. The technological level τ of these clusters covers practically the entire available range, i.e. the $[0,1]$ interval, with a non-trivial distribution. The clustering implies that communities of low level technologies can survive in the presence of highly advanced ones (see Fig. 6*(top)*). At $1 < r < 3$, where more advanced technologies are favored by the agents (locally the second largest τ), the cellular automata simulation of the system converges into an almost completely homogeneous state of a relatively high technological level (see Fig. 6*(middle)* where the specific case of $r = 2$ is plotted). In the simulations, initially clusters of agents with identical τ grow and finally the entire system evolves into a homogeneous state with all agents adopting the same technology. Since locally the agents choose the second highest τ to adopt, both very low and very high level technologies disappear during the evolution. The colors also illustrate that the limiting value of τ adopted by almost all agents is smaller than the highest available value $\tau_{max} = 1$, namely, it falls between 0.8 and 1. It follows from Eq. (10) that to reach the most advanced technologies, r has to surpass the threshold value $r = 3$ in the case of constant $n = 4$ number of neighbors. This regime is illustrated in Fig. 6*(bottom)* for the specific case of $r = 4$, where the light color in the last snapshots indicates that the most advanced technology $\tau_{max} = 1$ spraw onto the entire lattice.

5.1 Distribution of technological levels in cellular automata – extreme order statistics

In order to give a quantitative characterization of the time evolution of the cellular automata on a square lattice, we determined the distribution $p_t(\tau)$ of technological levels τ, and the mean $\langle \tau^t \rangle$ of technological levels at different times t

$$\langle \tau^t \rangle = \frac{1}{N} \sum_{i=1}^{N} \tau_i^t, \tag{24}$$

where N denotes the total number of agents.

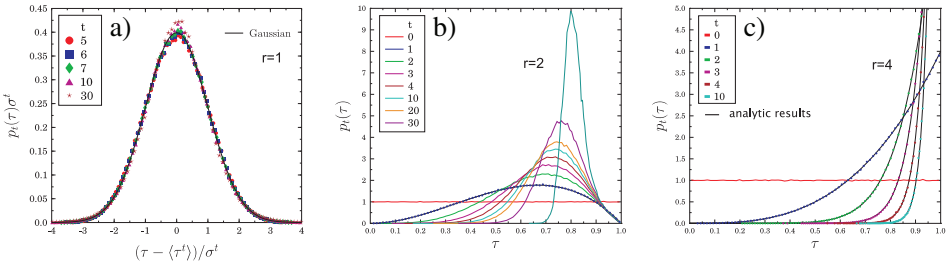


Fig. 7. Probability distribution of technological levels $p(\tau)$ obtained at different time values for three different values of the cost factor r. (a) For the case of $r = 1$ the rescaled plot of the distributions is presented, i.e. the distributions obtained at different times are rescaled by the mean $\langle \tau^t \rangle$ and by the standard deviation σ^t of the distributions. The master curve obtained perfectly agrees with the standard Gaussian. (b) For $r = 2$ when the second largest technological level is favored locally by agents, the distributions converge to a highly peaked functional form. (c) In the range $r > 3$ agents always select the most advanced technology, hence, the distributions can be very well described by extreme order statistics represented by the continuous lines.

Figure 7(a) shows that for $r = 1$, when higher level technologies do not provide advantages for agents, the distribution $p_t(\tau)$ rapidly attains a Gaussian shape. In order to demonstrate the validity of the Gaussian form, we present the rescaled distributions in the figure: the distributions obtained at different times are rescaled by the average technological level $\langle \tau^t \rangle$ of the corresponding state and by the standard deviation of the distribution σ^t so that $p_t(\tau)\sigma^t$ is plotted as a function of $(\tau - \langle \tau^t \rangle)/\sigma^t$. The high quality data collapse that can be seen in Fig. 7(a) and the good quality fit with the standard Gaussian

$$g(x) = 1/\sqrt{2\pi} exp(-x^2/2) \tag{25}$$

demonstrate the validity of the Gaussian description of the evolution of the cellular automata. The convergence to the Gaussian is very fast. In our test simulations after $30 - 40$ iteration steps the system completely forgot its initial uniform state and p_t attained the Gaussian limit distribution. This form implies that the fraction of agents having very high and very low level technologies both decrease and agents tend to copy technologies in the vicinity of the distribution mean. Consequently, the system does not have any technological progress, the average technological level remains nearly constant during the time evolution, and $\langle \tau^t \rangle \to 0.5$.

For the cost factor $r > 1$ agents locally prefer to adopt higher level technologies, namely, the highest or the second highest τ value of the neighborhood will be adopted on the square lattice

depending on the exact value of r. These local changes imply that the cellular automata rule Eq. (23) gives rise to a more complex time evolution involving also extreme order statistics. For $1 < r < 3$ all the agents adopt the second highest available technology; hence, in a large enough system the distribution of technological levels right after the first iteration step $p_t(\tau)$ can be described as the $k = 3$ rank extreme distribution Φ_M^k of $M = 4$ variables which are all sampled from a uniform distribution. In general, the probability density function $\Phi_M^k(x)$ of choosing the kth largest value among M realizations of the random variable x which has a probability density $p(x)$ and a distribution function $P(x)$ can be cast into the form

$$\Phi_M^k(x) = \frac{M!}{(k-1)!(M-1)!} P(x)^{k-1}(1-P(x))^{M-k} p(x). \tag{26}$$

It can be seen in Fig. 7(b) that by substituting the initial uniform distribution Eq. (21) into Eq. (26) with the parameter setting $M = 4$ and $k = 3$, a perfect agreement is obtained between our analytical predictions Φ_4^3 and $p_1(\tau)$ obtained from CA simulations. Unfortunately, at higher iteration steps the distributions p_t do not follow the functional form Eq. (26) when we substitute Φ_M^k and the corresponding distribution function recursively on the right-hand side. The reason is the overlap of the local neighborhoods of the lattice sites which modifies the statistics of technological levels. By increasing the number of iterations, p_t gets narrower and converges to a sharply peaked function as the final homogeneous state is approached (see Fig. 7(b) and Fig. 6). Consequently, the average value of the technological levels increases and converges to a limit value which is lower than the available maximum $\tau_{max} = 1$. It has to be emphasized that under this parameter settings the system exhibits considerable technological progress due to the disappearance of low level technologies and to the proliferation of the more advanced ones.

In the extreme case when the control parameter r becomes larger than 3, more advanced technologies provide so much benefit that it is always advantageous for agents to adopt the highest available technological level in the local neighborhood. Thus, $p_t(\tau)$ rapidly converges to a sharply peaked form the position of which approaches $\tau_{max} = 1$ through extreme order distributions (see Fig. 7c). It is interesting to note that contrary to the previous case of $1 < r < 3$, in this regime $r > 3$ the distribution p_t can be described by the extreme order density function Φ_M^k Eq. (26) with $k = M$ at any time t by taking into account that the size of the neighborhood M increases as a function of time t. We found a recursive formula for the time dependence of the parameter M

$$M_{t+1} = M_1 + 5 + 2(t-1), \qquad \text{with} \qquad M_1 = 4, \tag{27}$$

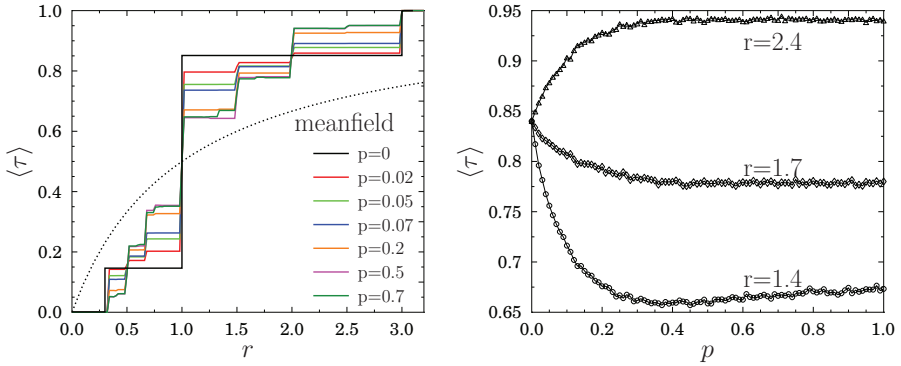
which shows how information spreads in the system. The lines in Fig. 7(c) demonstrate the excellent agreement of the above analytic prediction with the numerically obtained distribution functions. Note that due to the symmetry of the system with respect to the parameter value $r = 1$, the same holds also for $r < 1/3$ with Φ_M^1, where the smallest value $(k = 1)$ of M_t variables given by Eq. (27) is selected. These results imply that the average technological level in these regimes can easily be obtained analytically, i.e. the average of the largest and of the smallest value of M_t variables with uniform distribution can be cast into the form

$$\langle \tau_{max} \rangle = \frac{M_t}{M_t + 1}, \qquad \langle \tau_{min} \rangle = \frac{1}{M_t + 1}. \tag{28}$$

Substituting the recursive formula of M_t into Eq. (28) a perfect agreement is obtained with the numerical results of $\langle \tau^t \rangle$ Kun et al. (2007).

6. Agents on complex networks

The success or failure of newly introduced technologies on the market can also depend on the complexity of social contacts of users. This is especially valid for so-called networking technologies where the practical value of the technology for an agent depends on the number of social partners which already use the technology. Telecommunication technologies we are also focusing on are prototypical examples where the topology of social contacts may play a crucial role Mahajan & Peterson (1985); Rogers (1962).



(a) (b)

Fig. 8. (a) Average technological level $\langle\tau\rangle$ of the system obtained after long time evolution as a function of r for several different values of the rewiring probability p. (b) $\langle\tau\rangle$ as a function of the rewiring probability p for three specific values of r. The presence of long range connections can increase and even decrease the average technological level of the system depending on the cost factor r.

In order to have a more quantitative understanding of this phenomenon in the framework of our model, we implemented cellular automata simulations on a complex network of agents obtained by the Watts-Strogatz method varying the value of the rewiring probability p between 0 and 1 (see Section 3.3). The complexity of the underlying social network of agents introduces two important features: (i) as the rewiring probability p increases, more and more connections are established between remote agents introducing long range correlations in the system and reducing the "diameter" of the network. (ii) The probability distribution of the number of connections, i.e. the degree distribution of agents gets broader so that the number of connections can span from 1 to high values. One of the outcomes of the master equation approach was that the scatter of the degree makes the system more sensitive to the advantages technologies provide so in CA simulations more interesting details can be expected.
To go beyond the limitations of the analytic approaches we carried out cellular automata simulations on different topologies to determine the average technological level $\langle\tau\rangle$ in the final state of the time evolution when a frozen configuration is attained. The average technological level of the final state $\langle\tau\rangle$ is presented in Fig. 8(a) as a function of r for several different values of the rewiring probability p. Note that the average technological level $\langle\tau\rangle(r)$ is a

monotonically increasing function of r for any value of the rewiring probability p. The plotted functions are composed of distinct steps whose height and number are sensitive to the details of the network topology. The steps are the consequence of the behavior described by Eq. (9), i.e. the steps mark the interval borders for different degrees: e.g. for agents of $n = 4$ social contacts we have 4 intervals – taking also into account the symmetric cases of $r < 1$ and $1 < r < 3$ as well (see Eq. (9)) – which result in 3 steps. Increasing the rewiring probability p, the degree distribution $\rho(n)$ gets broader giving rise to an increase in the number of different degrees in the network which then results in a higher number of steps of $\langle\tau\rangle\,(r)$. It can be seen in the degree distribution of a rewired lattice of rewiring probability $p = 0.05$ in Fig. $2(b)$ that in this case the possible degrees of the network are $n = 2, 3, 4, 5, 6$. Using Eqs. (9) and (10) one can determine the interval limits of r for each n value, from which the overall r limits of the entire network can be obtained as $1/5, 1/4, 1/3, 1/2, 2/3, 1, 3/2, 2, 3, 4, 5$. For comparison, in Fig. $8(a)$ we also present the mean field solution Eq. (6) of the model obtained analytically for the fully connected case, when all agents are connected with all others.

A very important outcome of the above calculations is that the degree polydispersity of agents' social contacts makes the socio-economic system more sensitive to the details of the novel technology, i.e. to the specific value of the cost factor r. It can be observed in Fig. $8(a)$ that increasing the connectivity of the system, the presence of long range connections can increase but can also decrease the average technological level attained in the final state depending on the value of the cost factor r. For high enough cost factor r the long range contacts facilitate the spreading of advanced technologies, while for lower r values the opposite effect occurs, *i.e.* the dominance of low level technologies enhanced also by the long range contacts prevents technological advancement. Figure $8(b)$ provides some quantitative insight into this effect, where we present $\langle\tau\rangle$ as a function of the rewiring probability p for three different values of r. All the curves start from the same point at $p = 0$, since on a regular square lattice always the third highest technology is selected when r falls in the interval $1 < r < 3$. For increasing p the curves converge to r dependent asymptotic values which can be both lower and higher than the one at $p = 0$.

7. Discussion

In this chapter we presented an agent based model of the spreading of technological advancements, where the technology is used for the interaction/communication of agents. The model realizes a bottom-up approach to socio-economic systems which is especially designed for a cellular automata reprezentation. Agents/cells of the model can represent individuals or firms which use different level technologies to collaborate with each other. Costs arise due to the incompatibility of technological levels measuring the degree of difficulties in the usage of technologies. Agents can reduce their costs by adopting the technologies of their interacting partners. We showed by analytic calculations and computer simulations that the local adaptation-rejection mechanism of technologies results in a complex time evolution accompanied by microscopic rearrangements of technologies with the possibility of technological progress on the macro-level.

As a first step, simplified configurations of the model system were analyzed by analytical calculations: A mean field approach was considered where each agent communicates with all other agents. As to the next a master equation was derived which describes the discrete time evolution of the system assuming no spatial correlation, i.e. no clustering of agents according to their technological level. Already these simplified approaches revealed that the

rejection-adoption rule of our cellular automata leads to a homogeneous final state whose stability depends on the relative amount of advantages technologies provide for the users. These results also reflected the phenomenon that competition in the market takes much longer and it can be much more violent if the actors of the competition start from a nearly balanced initial scenario.

The analysis of the model system in its entire complexity was carried out by cellular automata simulations performed on a square lattice and on complex network topologies of social contacts. Computer simulations revealed that agents tend to form clusters of equal technological levels. If higher level technologies provide advantages for agents, the system evolves to a homogeneous state but clusters show a power law size distribution for intermediate times. The redistribution of technological levels involves extreme order statistics leading to an overall technological progress of the system. We also demonstrated that the topology of agents' social contacts plays a crucial role in the spreading process leading to a broad spectrum of novel behaviors. Analytical calculations and computer simulations showed that long range connections on the social network can facilitate but it can also hinder the diffusion of the advanced technology depending on the amount of advantages more advanced technologies provide with respect to the low level ones.

Our model emphasizes the importance of copying in the spreading of technological achievements and considers one of the simplest possible dynamical rule for the decision mechanism. In the model calculations no innovation was considered, i.e., agents could not improve their technological level by locally developing a new technology instead of only taking over the technology of others. Compared to opinion spreading models like the Sznajd-model Sznajd-Weron (2005); Sznajd-Weron & Sznajd (2000); Sznajd-Weron & Weron (2002) and its variants A.T. Bernardes (2002); Stauffer (2002a;b), the main difference is that in our case the technological level of agents is a continuous random variable; furthermore, the decision making is not a simple majority rule but involves a minimization procedure. Opinion of individuals can also be represented by a continuous real variable which makes possible to study under which conditions consensus, polarization or fragmentation of the system can occur. Such models show more similarities to our spreading model of technologies Gandica et al. (2010); Gilbert (2008); Hegselmann & Krause (2002). It is interesting to note that our model captures some of the key aspects of the spreading of telecommunication technologies, where for instance mobile phones of different technological levels are used by agents to communicate/interact with each other. In this case, for example, the incompatibility of MMS-capable mobile phones with the older SMS ones may motivate the owner to reject or adopt the dominating technology in his social neighborhood by taking into account the offers of provider companies of the interacting partners.

8. Acknowledgment

The work is supported by TÁMOP 4.2.1-08/1-2008-003 project. The project is implemented through the New Hungary Development Plan, co-financed by the European Social Fund and the European Regional Development Fund. F. Kun acknowledges the Bolyai Janos fellowship of the Hungarian Academy of Sciences. The authors are grateful for the generous support of Toyota Central R&D Labs., Aichi, Japan.

9.References

Albert, R. & Barabasi, A.-L. (2002). Statistical mechanics of complex networks, *Rev. Mod. Phys* 74: 47.

A.T. Bernardes, D. Stauffer, J. K. (2002). Election results and the sznajd model on barabási network, *Eur. Phys. J. B* 25: 123.

Castellano, C., Fortunato, S. & Lorento, V. (2009). Statistical physics of social dynamics, *Reviews of modern physics* 81: 591–646.

Gandica, Y., del Castillo-Mussot, M., Vazquez, G. J. & Rojas, S. (2010). Continuous opinion model in small-world directed networks, *Physica A: Statistical Mechanics and its Applications* 389: 218.

Gardner, M. (1970). Mathematical games – the fantastic combinations of john conway's new solitaire game "life", *Sci. Am.* pp. 120–123.

Gilbert, N. (2008). *Agent-based models*, Sage Publications, London.

Grassberger, P. (1984). Chaos and diffusion in determinicstic cellular automata, *Physica D: Nonlinear Phenomena* 10: 52–58.

Guardiola, X., Díaz-Guilera, A., Pérez, C. J., Arenas, A. & Llas, M. (2002). Modeling diffusion of innovations in a social network, *Phys. Rev. E* 66: 026121.

Hegselmann, R. & Krause, U. (2002). Opinion dynamics and bounded confidence: models, analysis and simulation, *J. Art. Soc. Soc. Sim.* 5: 3.

Helbing, D. (2009). Managing complexity in socio-economic systems, *European Review* 17: 423–438.

Helbing, D., Treiber, M. & Saam, N. J. (2005). Analytical investigation of innovation dynamics considering stochasticity in the evaluation of fitness, *Phys. Rev. E* 71: 067101.

Kocsis, G. & Kun, F. (2008). The effect of network topologies on the spreading of technological developments, *J. Stat. Mech* P10014.

Kun, F., Kocsis, G. & Farkas, J. (2007). Cellular automata for the spreading of technologies in socio-economic systems, *Physica A: Statistical Mechanics and its Applications* pp. 660–670.

Llas, M., Gleiser, P. M., López, J. M. & Díaz-Guilera, A. (2003). Nonequilibrium phase transition in a model for the propagation of innovations among economic agents, *Phys. Rev. E* 68: 066101.

Mahajan, V. & Peterson, R. A. (1985). *Models for Innovation Diffusion*, Sage Publications, London.

Rogers, E. M. (1962). *Diffusion of Innovations – first edition*, The Free Press, New York.

Sornette, D. (2000). *Critical Phenomena in Natural Sciences – second edition*, Cambridge Springer, Berlin.

Stauffer, D. (2002a). Monte carlo simulations of sznajd models, *J. Artif. Soc. Soc. Simulation* 5: 1.

Stauffer, D. (2002b). Sociophysics: the sznajd model and its applications, *Int. J. Mod. Phys C* 13: 315.

Sznajd-Weron, K. (2005). Sznajd model and its applications, *Acta Phys. Pol. B* 36: 2537.

Sznajd-Weron, K. & Sznajd, J. (2000). Sociophysics: the sznajd model and its applications, *Int. J. Mod. Phys C* 11: 1157.

Sznajd-Weron, K. & Weron, R. (2002). A simple model of price formation, *Int. J. Mod. Phys C* 13: 115.

von Neumann, J. (1948). The general and logical theory of automata, *L.A. Jeffress (Ed.). Cerebral Mechanisms in Behavior* pp. 1–41.

Watts, D. J. (1999). Social percolation, *American Journal of Sociology* 105: 493.
Watts, D. J. & Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks, *Nature* 393: 440–442.
Weidlich, W. (2000). *Sociodynamics*, Dover Publications, MIneola, USA.
Wolfram, S. (2002). *A New Kind of Science*, Wolfram Media, Inc.

17

Cellular Automata based Artificial Financial Market

Jingyuan Ding
Shanghai University
China

1. Introduction

Rational investor hypothesis, efficient markets hypothesis(EMH), and random walk of yield rate are three basic concepts of modern capital market theory. However, it could not be proved that real capital markets are full with rational investors. The theory, which regards the price movement of capital market as random walks, and regards the yield time series as a normal distribution, is not supported by the real statistics data usually. A capital market, in essence, could be regarded as a complex system, which consists of masses of investors. Investors make investment decision basing on the public or private information inside or outside the market. The movement of price and volume is the emergency of investors' group behavior. With the sustained growth of computational capabilities and the appearance of complexity science, artificial life, multi-agent system (MAS), and cellular automata (CA) are provided for the modeling of complex system. Researchers got powerful tools to build discrete dynamics model for the capital market for the first time. The Santa Fe artificial stock market(SF-ASM), which is presented by Santa Fe institution in 1970s, is the original version of the artificial financial market(AFM). Modeling for the microstructure of the capital market, made the verification and falsification of economics theories possible. On the part of macroscopic statistical data of the market, a series non-linear dynamic analysis method, such as fractal statistics, had been applied to analysis of financial time series. New research methods, which are used both in microscopic and macroscopic aspects of capital market, help us build brand new dynamic models for capital markets.

The appearance of SF-ASM has influenced this area deeply. Most successors are the variety or improvement of SF-ASM. SF-ASM is a kind of MAS, which focuses on simulating heterogeneous investors' investment behaviours. In my opinion, the investment process of an investor can be divided into 2 steps: forecasting and decision. The forecasting step is how an investor considers public or private information inside or outside of the market. And the decision step is how an investor reacts to the prediction. Rational investor hypothesis and various investment decision processes in SF-ASM are just different ways to deal with information. Basing on neoclassicism economics, EMH announce that the price in the market reflects all information, or at least all public information, and that rational investors react to these information in the same way. Multi-Agent based SF-ASM supports heterogeneous investors in reacting to information in various ways, but provides public price as the only information. The fact that information relating to the market is homogeneous and public to each investor can be compared to the gas filling the whole

"container of market". However, as we know, in real capital markets, except public information including the announcement, annual report, interest rate etc., there are also inside information, individual attitudes or predictions, and even emotions, which can influence investors' investment. What's more, information is time sensitive. Because non-public information may reach investors in different time, the situation of real capital market could be more complex. So SF-ASM is more "efficient" than real capital markets for it's simplifying the description of information.

If we describe the non-public information in an AFM model, the interoperation among individual investors can be expressed certainly. As a result, the cellular automaton (CA) is adopted. Classical CA is a kind of large scale discrete dynamical systems. Each cell in CA can interoperate with neighbors in a local scope, which is defined by CA's neighborhood. Yi-ming Wei, Shang-jun Ying, Ying Fan, and Bing-Hong Wang presented a CA based AFM in 2003. In this model, the local interoperation of CA is used to describe the spread of the herd behavior in capital markets. However, the neighborhood of this CA based AFM is still classical Moore neighborhood. All the investors in this AFM have the same simple investment behavior rule. The pricing mechanism of the market is far from the realistic markets. In real capital markets, as we know, the non-public information spreads through the investors' social network, rather than 2-D lattice. The connectivity, diameter, and degree distribution of the social network can decide the speed and scope of the information spreading. Furthermore, social network is not a fix, but dynamic structure.

According to the above reasons, combining the feature of multi-agent system and complex network, we extend the definition of CA in following aspects in this chapter: Neighborhood with network topology is adopted in CA; Structure of neighborhood is no more fixed, and will change following the neighborhood evolution rule; Cells in CA are no more homogeneous, and each cell has its own state transfer function with the same interoperation interface. Combining the above extensions of CA, as well as the other researchers' research on cellular automata on networks (or graph automata), we present a formal definition of CA on networks.

On the basis of CA on networks, a new artificial financial market modeling framework, Emergency-AFM (E-AFM), is introduced in this chapter. E-AFM provides all standard interfaces and full functional components of AFM modeling. It includes classification and expression of information, uniform interfaces for investors' prediction and decision process, uniform interface for pricing mechanism, and analysis tools for time series.

E-AFM is a modeling framework for any kind of AFM. By instantiating the investors' asset structure, neighborhood network, behavior rules of investors, and pricing mechanism, we can get a specific AFM model. After an AFM model is simulated, we can get a price and volume time series in standard format just like real capital markets. Analysis tools provided by E-AFM, such as Hurst exponent and Lyapunov exponent, can be used to measure the fluctuation feature of price/yield time series. We can compare the simulation data with the real capital market data. Also we can find the relationship between the fluctuation feature and the topology of social networks.

In the rest of this chapter, an E-AFM based AFM model is introduced. This model is a simple model which is designed to find the relationship between the fluctuation feature of price time series and the degree distribution of the social network (neighborhood of CA). The statistics feature of neighbourhood structure is observed and compared with the fluctuation feature of price/yield time series. It is not a perfect model to get a new capital theory, but we can still realize how cellular automata can help us to do research in financial area.

2. The capital market in viewpoint of complex system

The capital market has existed for hundreds of years. And it is one of the most essential part of modern societies. However, people know little about capital market till today, even through which influences everyone's benifit. There is still no capital market theory which can explain the inner dynamic mechanism of capital market strictly. The capital market is one of the complex systems, which created by human being self, but are difficult to understand by us. Traditionally, when we discribe a uncertain system, we consider it a stochastic system. For example, modern capital market theory is based on probability statistics theory. Actually, however, there are many strict conditions for stochastic systems, such as, independence assumption. So, it is not rigorous to classify any uncertain behaviour to stochastic system. In order to apply stochastic process tools into modern capital market theory, its founder made many strong assumptions, such as, Rational investor hypothesis, efficient markets hypothesis(EMH), and random walk of yield rate. Unfortunately, neither these assumptions could be supported by investor psychology and behaviour analysis, nor their conclusions could be proved by market statistics data. If we investigate the capital market in viewpoint of complex system, we can find that the complexity of this system's behaviour is never as simple as random walk, but comes from extremely complex internal structure of capital market. We need some approaches to find out what assumptions are reasonable, what caused the fluctuation in price, and what theory is reliable and verifiable.

The complexity of system can be classified into time and space complexity. That is to say, we can investigate it in behaviour and structure aspects. From the standpoint of time, some extremely simple system, such as nonlinear dynamic systems like Logistic equation, can present extremely complex dynamic behaviour. This kind of dynamic systems, however, which have explicit equations, could be investigated in mathematical methods. The degree of freedom of this kind of system is finite and knowable. And their behaviours are still reproduceable in controlled conditions. The behaviour complexity of a real complex system comes from its structure complexity. What we call structure complexity means that the degree of freedom is too complex to reproduce its dynamic feature in classic analytical way. The structure complex could be reflected in the uncertainty of degrees of freedom, as well as in the interdependence of the components. The Name of two books: "Hidden Order: How Adaptation Builds Complexity" (Holland, 1996) and "Emergence: From Chaos To Order "(Holland, 1999), are good summary of the formation of the complex system. When individuals in a system interact with each other, their adaptive behaviours are the inner rules, or "hidden order", of system dynamics. Due to the quite huge amount of the individuals, and intricate interdependence within them, the whole system would represent some complex dynamic feature which could be observed by us. This process is called by John Holland "Emergence". John Holland's viewpoint explained how complex systems appear. The adaptive individuals are not organized in some regular or linear way. They don't act randomly and independently with each other either. The individuals with their autonomous targets in a system, may form some stable structures which are hard to know, during their adaptive behaviours. These stable structures make these complex system much harder to investigate than both absolutely ordered systems and absolutely disordered systems. John Holland calls these stable structures, which formed during adaptive interactive behaviours, "patterns". For a complex system, pattern is key to explore the relationship between microstructure and macrodynamics of the system.

From above discuss, we can conclude some essential conditions of an complex system:

- The system is composed of a large amount of individuals with their autonomous targets. An individual's target is the reason of its adaptive behaviour.
- Individuals in the system would interact with each other in a local scope. Interactions within individuals made the system an organic whole. The locality of these interaction is the condition of patterns in the system.
- The feature of the patterns decides the complexity of the system dynamics.

From our experiential knowledge about capital market, it satisfies above conditions exactly. The macrodynamics of capital market is price and volume movement. And the movement of transaction data comes from the trading orders quoted by masses of investors. Most investors participate in the capital market to earn profit. There still may be some investors with other targets. But at least all participants of capital market have their target. So capital market satisfies the first condition.

The investment decisions of investors are based on the predictions on the future price. Investors' predictions come from their judgement on different kinds of informations, such as macro-economy policy, profitability of the company, history transaction data, important news, influences from other investors etc. Some kinds of informations are public informations, the others spread through the investors' interactions. The interactions within investors are direct and local, just like other kinds of social networks. Capital market satisfies the second condition too.

An capital market is comprised of masses of investors with heterogeneous features, which involving the investor's condition of assets, information source, and risk preference etc. We call all about these "market structure". Different market structure can decide the complexity of macrodynamics of capital market. This matches the third condition of complex system. Actually, different hypotheses about market structures decide different capital market theories. For example, the rational investor hypothesis assumes that investors are seeking effectiveness of mean/variance in Markowitz meaning; efficient markets hypothesis assumes that investors in capital market can get infinite risk-free credit, which means investors can buy or sell as long as they wish; and only public informations, which had been reflected in market price already, can influence investors' decisions. In this kind of market structure, the dynamic feature of market price or yield is a random walk. In later sections of the chapter, we can see more models, in which market structure decides the feature of price fluctuation.

According to above discuss, we consider that treating capital market as an complex system is reasonable to explore its dynamic mechanism. Building models for an complex system is the best way to research it. Neoclassic financial theory can be treated as a kind of system model of capital market without direct interaction within investors. The subsequent theories, such as Coherent Market Hypothesis (CMH) or Fractal Market Hypothesis (FMH), could be treated as other kinds of capital market models which emphasize heterogeneity and direct interactions of investors. When we build models for complex system, we just can design interaction rules according to our experience, logical reasoning, or conclusions from psychology and behavioristics. If we want to verify the rationality and correctness of a model, we must evolute it and compare the macrodynamics of the model with the real system. Fortunately, masses of transaction data had been accumulated in real capital markets, and there are some effective methods to analyse time series. The conditions to build a verification system for capital market theories are equipped now. The approaches to verify capital market theories are usually collectively called Experimental Finance.

3. Introduction to previous works on artificial financial market

In the development of complexity science, some modeling tools such as cellular automata and muti-agent system appeared. In the 1980s, because of the influence of artificial life (Christopher Langton, 1986), ideas like complexity, evolution, self-organization, and emergence are applied into the modeling of social system. Researchers in Santa Fe Institute first introduced Agent-based Computational Economics into financial area. Their Santa Fe Institute Artificial Stock Market(SFI-ASM) was the pioneer of artificial financial market. In recent years, group behaviours in capital market attract many researchers, and cellular automata was introduced to build artificial financial market models.

3.1 Senta Fe Institute Artificial Stock Market

A classic Santa Fe Institute Artificial Stock Market(SFI-ASM) includes N interactive agents, and a stock market, or an exchange, which is avialable to perform stock exchange. Agents in SFI-ASM could belong to different categories, or in a sense, they are heterogeneous. There is not direct interaction within agents in SFI-ASM. They just interact with each other through trading in the exchange. Time in SFI-ASM is discrete. Period t lasts from time t to $t+1$. At end of each period, bonus would be allocated to each share, following time series $d(t+1)$. The bonus time series is a stochastic process, which is independent of the stock market or the agents. Ornstein-Uhlenbeck process is often adopted as the bonus time series. There are even a fixed-income asset with fixed interest rate r, such as bank, in the market. Agents can decide invest how much money into the stock market or left it in bank. At any time t, *agent i* holds some shares of stock $h_i(t)$, and lefts a part of cash in bank $M_i(t)$, then the total assets of *agent i* is:

$$w_i(t) = M_i(t) + h_i(t)p(t) \tag{1}$$

Where $p(t)$ is the price at time t. After a time step, the value of the asset portfolio is:

$$\overline{w}_i(t+1) = (1+r)M_i(t) + h_i(t)p(t+1) + h_i(t)d(t+1) \tag{2}$$

Note: $\overline{w}_i(t+1)$is not $w_i(t+1)$, $\overline{w}_i(t+1)$ does not includes transaction in next time step, which could cause changes of the cash in bank or the shares held by the agent.

All agents in SFI-ASM have the same utility function and risk preference. But each agent has a condition-forecast rule itself. The form of condition-forecast rule is as follows:

if (condition fulfilled), then (derive forecast).

It can be seen that it is a general form for condition-forecast rules. Different agents can use different rules, such as basic analysis or technical analysis, to predict trend of future price. And then, agents can make invest decision based on the predictions. So the agents in SFI-ASM are heterogeneous fundamentally. Artificial intelligence methods like artificial neural network and genetic algorithm can be applied to condition-forecast rules to provide agents self-learning and self-adaption abilities.

In SFI-ASM, there is a "specialist" who controls the Trading Process. He decides the price of next time step ($p(t+1)$) according to supply and demand in the market. When in oversupply, the price would drop, when in short supply, the price would rise. The specialist influences the investment decision of agents by the fluctuation in prices. Price, bonus, the size of all bid

and ask orders, compose the global information variable of SFI-ASM. The global information variable is the foundation of agents' prediction of next time step.
SFI-ASM is the pioneer of artificial financial market and has revolutionary influence on experimental finance. Many models derived from SFI-ASM appeared in its long-term development. Creators of SFI-ASM introduced the methodology of complex system modeling into financial area. In SFI-ASM, investors are regarded as initiative individuals (agents) in the system. The investors' investment behaviours on the market are regarded as the interaction within them. The investment behaviour of an investor was divided into three stages: price prediction, making judgement by utility function, makeing investment decision according to risk preference. The heterogeneity of agents is reflected in the price prediction stage. The other two stages keep homogeneous.
However, some shortcomings of SFI-ASM come into our notice. Information is vital important to real capital market. Because all investment behaviours are based on predictions, and information is the fundamental of predictions. As we know, the basic viewpoint of efficient markets hypothesis is that the price had reflected all information related to the market. The sceptics of efficient markets hypothesis queried this assumption greatly. In real market, the spread of information is complex. There is public global information which can reach all investors at the same time. Such as trading data, financial policy, news of the company are this kind of information. There is also non public information in the market. The personal viewpoints, emotion, insider information, are just trasfered from individual to individual in a local area. Some individuals respond the information immediately after they receive it; others may just wait till the information is verified. The delayed responses may cause more complex phenomenon in capital market. As the information is time sensitive, different investors with different "investment start point"(Peters, 1996) would be interested in different kinds of information.
The SFI-ASM, which based on muti-agent system, focuses on heterogeneous investment behaviours of investors. But only simple, public, global information without delay, was adopted in SFI-ASM. In SFI-ASM, information spreads in the market at the same time, and would be handled by agents immediately. Excessive simplification of information and ignoring local direct interaction may make SFI-ASM close to efficient markets hypothesis. Or in other words, the complexity of SFI-ASM comes from complex individual behaviours, other than inner structure formed by self-organization of individuals.

3.2 The classic cellular automata based capital market model

Recent years, the non public information's influence on capital market came into researchers' sights. Especially, under some specific culture environments or the market is not developed or mature enough, public information is not transparent or reliable, non public personal information could be decisive. Group psychology and herd behaviour appeared frequently in the emerging market like Chinese capital market.
It is necessary to describe the interaction within individuals if we want to build a model for the spread of non public information. Cellular automata is superior in this aspect. The classic cellular automaton is kind of discrete dynamic system which is composed with masses of individuals. The behaviour rule of the individuals is simple and unique in a cellular automaton. The interactions within individuals rely on neighbourhood structure in the cellular automata. These features can be used to express direct non public information exchange within investors.

One of the typical cellular automata based artificial financial markets is "the cellular automaton model of investment behavior in the stock market"(Wei et al., 2003). In this model, stock market is regarded as a cellular automaton. And the investors are regarded as cells in 2D lattice space. The neighbourhood of a cell follows the Moore's definition (Fig. 1.).

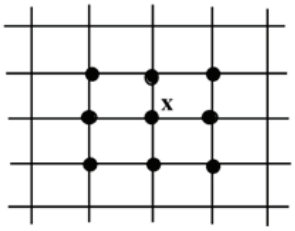


Fig. 1. Moore's neighbourhood

"The cellular automaton model of investment behavior in the stock market" focused on the influence of herd behaviour on the capital market. In the model, a cell have just three states (attitude): buying, holding and selling. In this model, the unit of time is step. At step *t*, a cell's state would be decided by states of neighbours at step *t-1*, according to its state transition rule. The state transition rule would calculate the distribution of buying, selling, holding neighbours, and decide the state at step *t* itself. In each step, the model would figure out a price according to the distribution of cells' states.

Compare with SFI-ASM, "The cellular automaton model of investment behavior in the stock market" has totally different standpoint about market information. In this model, only local information inside the neighbourhood can influence a cell's investment decision. No public information is taken into account. The primary importance of "The cellular automaton model of investment behavior in the stock market" lies in introducing the local interaction of investors into capital market models, and comparing the relativity between group psychology and VAR(Value-at-Risk). However, this model just focused on group psychology in the market, ignored all other factors involved with price fluctuation. Its pricing mechanism is too subjective, and it is much less mature than SFI-ASM. Even regarding the interaction within cells, the 2D lattice space and Moore's neighbourhood definition are not suitable for social relationship. Actually, social relationship is usually a network. Its structure influence the spread dynamic feature deeply.

4. The formal definition of cellular automata based artificial financial market

As discussed above, the capital market is a dynamic system with masses of individuals interacting with each other. Individuals have their own target. The behaviour of an individual relys on information based prediction. In essential, the difference in different capital market theories and models lies in different standpoint about information's category, spread, and handling. Further more, we consider that the complexity of the capital market dynamic, comes from the inner structure which is formed in the process of the individuals' self-organization. Both muti-agent and cellular automata are suitable for modeling of capital market. As there is neighbourhood definition in cellular automata to limit the interaction scope of cells, it is superior in describing non public information in capital market. If we extend the definition of classic cellular automata, make it can contain heterogeneous cells

and social relationship neighbourhood, it would be a better choice to build artificial financial market based on cellular automata. Before we can do so, it is necessary to extend classic cellular automata in some aspects.

A *d*-dimensional classic cellular automaton could be defined as a quadri-tuple:

$$\wedge = \left(Z^d, S, N, \delta\right) \tag{3}$$

Z^d stands for a *d*-dimensional discrete lattice space. It's the space structure of *d*-dimensional classic cellular automata.

S is the finite states set of cells.

$N=\{n_j = (x_{1j}, \ldots, x_{dj}),\ j \in \{1, \ldots, n\}\}$ is the finite ordered subset of Z^d. *N* is called the neighbourhood of cellular automata.

$\delta: S^{n+1} \rightarrow S$ is the local state transition function of Λ.

We can find from this definition that the essential feature of cellular automata is its discrete space-time and local interaction. If we want to apply it to social system modeling, we must extend its definition in four aspects.

The cellular automata focus on how individuals' adaptive behaviors result in complexity of the system. But when we build some models for real world, there is public information which can influence individuals behaviours as well as interaction within them. If we adopted public information in cellular automata, it would become an open system.

Traditionally, cells in the cellular automata are homogeneous. That means all cells in a cellular automata have the same state transition function. But sometimes, we need to include individuals who would respond to the information in various way. The problem of heterogeneous cells is that cells must interact with neighbours who may have different state transition function. So we must guarantee the *S* in the quadri-tuple can be accepted by all cells' state transition function, even though they may have different logic.

The neighbourhood of cellular automata represents interaction scope of a cell. The space of social system is not like physical system. The relationship within social members is some kind of networks. So *d*-dimensional discrete lattice space must be replaced by network space. In fact, network is a universal description for discrete space. The *d*-dimensional discrete lattice space is just an example of it.

In classic cellular automata, the neighbourhood is fixed. In social system, however, the relationship between two members is not so stable. The adaptive behaviors of individuals are even the cause of formation of the system's inner structure. Margolus designed odd-even neighbourhood for odd-even steps, then realized the change of neighbourhood. In the cellular automata based 2-dimensional fluid model: HPP Lattice Gas Automata(Hardy et al., 1973), Margolus neighbourhood is adopted. The successor of HPP model: FHP Lattice Gas Automata (Frisch et al., 1986), change the lattice into hexagon. The neighbourhood of FHP model is alterable too. Network dynamics plays an increasingly important role in social networks modeling. We could add network dynamics as the neighbourhood transformation rule into the definition of cellular automata.

Considering the four extends, we can get a new definition for cellular automata:

$$\wedge = \left(Z, S, N, P, \delta, \sigma\right) \tag{4}$$

Because the public information and neighbourhood transformation function are supported in the cellular automata, the new definition becomes a six-tuple. In the new definition, *Z*

insteads the original Z^d, which means the space of cellular automata dosen't have to be Euclid space. It could be a network or graph structure. Accordingly the N in the cellular automata may follow the graph's neighbourhood definition. Further more, the N dosen't have to be stable. σ: $N \rightarrow N$ is the neighbourhood transformation function, which can change the neighbourhood in every evolution step of the cellular automata. P stands for the public information. Accordingly the state transition function becomes δ: $P, S^{n+1} \rightarrow S$.

Although we extended the definition from classic cellular automata, we still kept its essential features. The new kinds of cellular automata are still time-space discrete system. Each cell decides its state in next step according to the states of neighbours and itself in current step. The macrodynamics of cellular automata is the emergence of masses of cells' adaptive behaviours. The classic cellular automata could be regarded as an instance of the new definition. Because the cells could be heterogeneous, a cellular automaton under new definition could also be a multi-agent system.

New definition of cellular automata gives us a foundation to define a cellular automata based artificial financial market. Because there could be many artificial financial markets under different assumptions, we just define the general part of them. In a cellular automata based artificial financial market, each cell represents an investor. Z in the formula (4) is the set of cells. In cellular automata based artificial financial market, the finite states set S is a 6-tuple:

$$S = (C_u, C_f, S_u, S_f, Q, E) \tag{5}$$

C_u and C_f stand for usable and frozen cash respectively. S_u and S_f are usable and frozen stock respectively. Q is the set of orders which have been quoted to exchange house but have not been completed or canceled yet. Each order includes direction (ask or bid), price and amount. E, which valued rise, fall, or keeping, is the price prediction of an investor. C_p is the total property of an investor. Given P as current stock price, C_p can be expressed as follows:

$$C_p = C_u + C_f + P\left(S_u + S_f\right) \tag{6}$$

Maximization of C_p is the only goal for all investors.

N is the neighborhood in cellular automata. In this ASM, N is a directed graph. When $Cell_i$ is making a prediction, a directed edge <i, j> between $Cell_i$ and $Cell_j$ exists only if the $S.E$ value of $Cell_j$ can affect the $S.E$ value of $Cell_i$. In this condition, $Cell_j$ is defined as a neighbor of $Cell_i$. The neighbour relationship between these two cells is not self reciprocal.

In the cellular automata based artificial financial market, public information includes trading data, public financial policy, such as risk free rate, and company news, such as financial reports and bonus. As the definition of cellular automata, public information is represented by P. The state transition function decide a cell's state according to the public information and the states of the cell self and the neighbours. So is defined as:

$$\delta : P, S^{n+1} \rightarrow S \tag{7}$$

The neighborhood transition function is:

$$\sigma : N \rightarrow N \tag{8}$$

σ is the variance of the dependency relationship between cells. Based on different assumptions, methods to rebuild the dependency networks can be different. σ would be performed after each trading day.

Once we gave the extended definition of cellular automata specific meaning, we defined a cellular automata based artificial financial market. We try our best to abstract the essential of the capital market. We emphasize the heterogeneous individual behaviours, as well as the complex information spread in the market. We believe the information is the decisive factor for a predictive system. The Emergence-Artificial Financial Market Framework, which would be introduced later, is a realization of the cellular automata based artificial financial market.

5. The emergence-artificial financial market framework

Now we can build artificial financial market models under above definition. As we know, the target of artificial financial market is to find out the relationship between macrodynamics and microstructure of the capital market, and verify the financial theories which are based on different assumptions. These assumptions are focus on the investors' behaviours and the spread of information. Other components of the capital market are stable and clear. So, we built a framework, realized the common parts of the capital markets in it, and defined the interfaces of the heterogeneous investors and informations. Because the complex macrodynamics could be regarded as the emergence of the adaptive behaviours of individuals, we named the framework "The Emergence-Artificial Financial Market Framework (E-AFM)".

5.1 The structure of E-AFM

As we defined in section 4, an E-AFM is a cellular automata based artificial market, so, first, we realized a cellular automata library under the extended definition. Then we realized E-AFM as a template instantiation of the cellular automata library. All these frameworks are realized in C++ language, in order to utilize its generic programming mode and parallel technology.

The basic starting point of the cellular automata library is abstraction of the data type of the cell state (personal infermation) , and public information. That's why we use parameterized type feature of C++ template. The base classes of cell, cells' container, neighbourhood, are provided in the library. The base classes of cell and cells' container are both template classes. The template parameters are the abstract data types of cell state and public information. The template parameter StateType is the data type of the cells states. Users can define it according to their needs. In the CellBase<StateType> class, state transition function is declared as a pure virtual function, any class, which derived from CellBase<StateType>, should overwrite the state transition function in its own rule. Two derived classes of CellBase<StateType> were provided in this library, one is for synchronous cellular automata, and another is for asynchronous cellular automata. When we realize a cellular automaton, we just need to define the data type of personal and public information, design a class derived from class SynchCellBase<StateType> or class AsynchCellBase<StateType>, and provide relevant state transition function.

In the cellular automata library, all cells are managed by cell container classes. The design targets of the cell container include following aspects. Firstly, the cell container should provide one or more kinds of traversal methods to access all cells in the cellular automaton. Secondly the cells' random access should be supported, because we can't assume the structure of users' cellular automata, and we need to access a cell through its neighbors. Thirdly, the neighborhood of the cellular automata should have an inner expression in the container. That means, when we access a cell, it is required to get the cell's neighbors

directly. Lastly, both serial and parallel accesses must be supported by the container. In detail, when different threads access different cells without mutex at the same time, the container should be thread safe. When different threads access the same cell at the same time, a mutex would be provided.

In our cellular automata library, the solution to satisfy the requirement of concurrency is class concurrent_vector which is provided in Microsoft Concurrency Runtime technology. If we use this library to build a cellular automaton model, all container classes should derive from class CellContainerBase<StateType>, which uses a thread safe container class concurrent_vector to store cells. So we can access any cell randomly by its index. The class CellContainerBase<StateType> has a member pointer, which points to a derived class of class Neighborhood. Neighborhood class declared two basic abstract functions. The function *AppendItem* is used to add a new cell's index into the relationship structure. The function *Neighbors* is required to return a cell's neighbors' index. The derived classes of neighborhood are required to realize the two functions. The purpose of adopting index to manage cells' neighborhood is to separate the design of container class and neighborhood class. Two derived classes of CellContainerBase<StateType> are provided to perform the evolution of the cellular automata in serial or parallel way.

As discussed above, an artificial financial market would be regarded as a cellular automaton model. So, E-AFM, as a framework, realized the main parts of this kind of cellular automata. The realized parts include the investment process of each investor (cell); the interaction way with cells; the basic mechanism of the artificial financial market; the definitions of evolution step and trading day; the account management etc. But the basic assumptions to the capital market are left to model designers. For example, model designers can define the data type of the public or private information, decide the structure and the evolution of neighborhood, design the behavior of investors, and choose the price formation mechanism of the capital market, such as order-driven or quote-driven rule.

The E-AFM is a template instantiation of the cellular automata library. As discussed above, the state type of an investor has been defined clearly in formula(5). The classes which are related to cash account, position account, trading orders, and investor's attitude are provided in E-AFM as cell state. So we can instant the template classes of the cellular automata library, and provide their derived class in E-AFM. SimInvestorBase class is derived from CellBase<StateType> class. It provided functionality to manage cell state itself, but it is still an abstract class, because the investment process are left to users to realize. There are also some classes derived from neighbourhood class provided in E-AFM. These neighbourhood classes can change their structures after a trading day. Especially some neighborhood transition functions are related with individuals' state. There are other components in E-AFM, which provide stable functionalities such as account management, exchange, quoted order management, and pricing mechanism etc. It is not necessary to derive them usually.

When we simulate the artificial financial market, we are performing the evolution of the cellular automata. Utilizing the Concurrent Runtime technology, the simulation could be parallel. The benefits of parallel simulation are not only higher performance. It is more important that the concurrent evolution is more close to reality. A trading day was defined as an evolution with several steps in the E-AFM. After a trading day, the artificial financial market would be closed, and the accounts' settlement would be performed. The neighbourhood of the cellular automata could be rebuilt too. One simulation of an E-AFM instance could include hundreds of trading days.

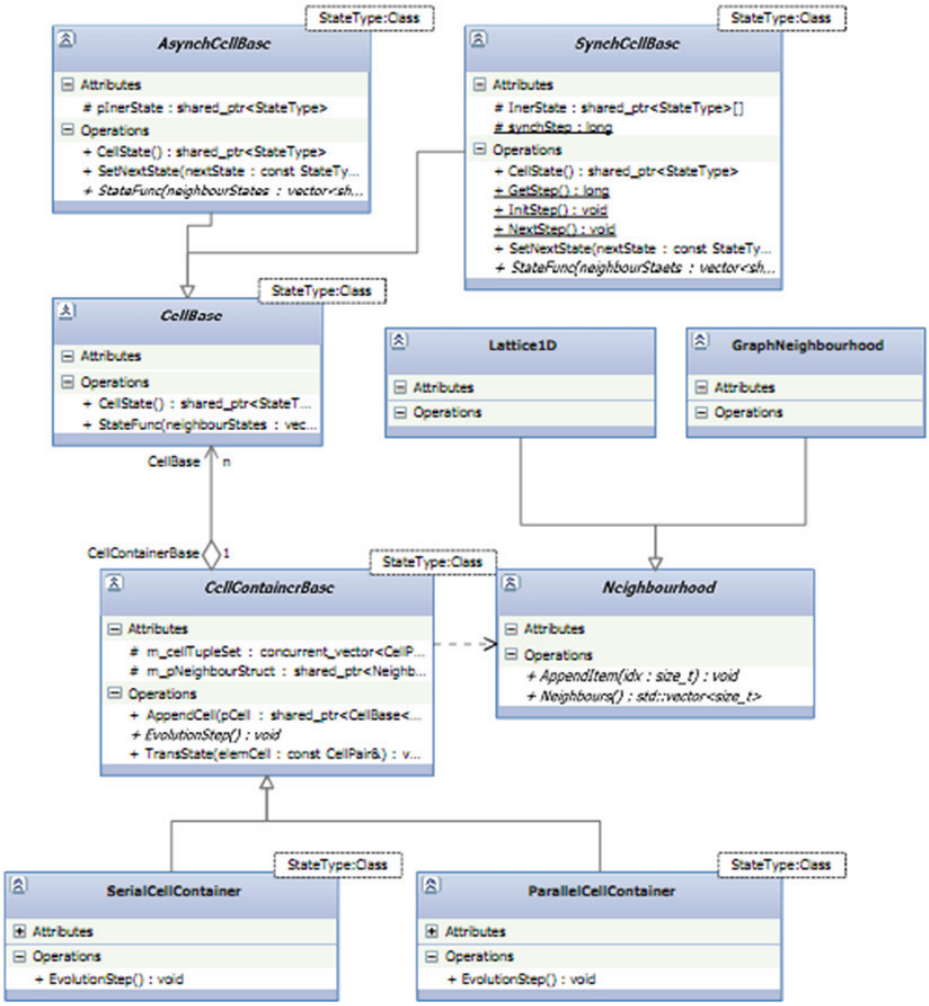


Fig. 2. Static Structure of Cellular Automata Library

5.2 Analysis tools in E-AFM

After a simulation finished, trading data would be produced as real capital market. Some tools are provided in E-AFM to analysis the macrodynamics and microstructure of cellular automata based artificial markets. Some of these tools can also be used to analysis trading data of real capital market.

One of the analysis tools is the Hurst exponent which was introduced into financial time series analysis first by Mandelbrot. Mandelbrot consider the Hurst exponent is better than variance analysis, spectral analysis, and autocorrelation. Hurst exponent is mainly used to estimate the long term memory of time series. R/S analysis (Rescaled Range Analysis)

(Hurst, 1951) is the most classic estimation method of Hurst exponent. Edgar E. Peters used R/S analysis to find the fractal feature of financial time series, and built the Fractal Market Hypothesis (FMH). R/S analysis is also provided in E-AFM.
We have a time series, which length is T. First, we should divided the time series into N adjacent v-length sub-periods, and $N*v=T$. Each sub-period is recorded as I_n, $n = 1,...,N$. Each element in I_n is recorded as $r_{t,n}$, $t = 1,2, ...,v$. M_n is the arithmetic mean value of I_n. We can calculate the the accululated deviation $X_{t,n}$ from the mean using following equation:

$$X_{t,n} = \sum_{u=1}^{t} X_u - M_n \tag{9}$$

Let:

$$R_n = \left(\max(X_{t,n}) - \min(X_{t,n})\right) \tag{10}$$

R_n is called range of I_n. Let S is the standard deviation of the I_n. Then the Rescaled Range is defined as:

$$E\left(R_n/S_n\right) = \left(aN\right)^H \text{ as } N \to \infty. \tag{11}$$

or:

$$\log\left(E\left(R_n/S_n\right)\right) = H\log\left(N\right) + \log\left(a\right) \tag{12}$$

The slope H is the Hurst exponent. It can be estimated by least square method or other methods.
One of E-AFM's tasks is to find how does the microstructure of the capital market cause complexity of macrodynamics. For example, the structure of neigbourhood graph can influence the spread of non-public information in the capital market. We use the degree distribution to measure the complexity of the networks, and use the clustering coefficient to measure the dependency level within the investors. The clustering coefficient γ_v of a vertex v in a graph can be defined as:

$$\gamma_v = \frac{\left|E\left(\Gamma_v\right)\right|}{\binom{k_v}{2}} \tag{13}$$

Where Γ_v is the neighbourhood of vertex v, and $|E(\Gamma_v)|$ is the number of edges in the neighbourhood. $\binom{k_v}{2}$ is the maximum number of possible edges in the neighbourhood.
There are still many other methods could be used to measure the time-space complexity of the artificial financial market. Due to space limitations, we don't discuss them individually.

6. Dynamic analysis of an Artificial Financial Market

In the last part of this chapter, we'd like to show the readers an example artificial financial market which is based on E-AFM, and analysis the results of simulation. The assumptions of this artificial financial market are not complete enough to proclaim a new capital market

theory. But it can show us that the cellular automata based artificial financial market could be an effective tool to simulate the process of self-organization in the capital market. And it can also show how does the structure of the social network influence the spread of information and then influence the price fluctuation.

In this artificial financial market, we assume that an investor's behaviour can be divided into prediction stage and investment stage. In the prediction stage, the investor predicts the direction future market price (tuple E in equation 5) according to the public and non-public information. The public information is the technical analysis on history trading data, such as moving average convergence/divergence index (MACD). The non-public information is the collection of neighbours' attitudes. Each investor has a weight number δ_i decides the different influence of the public and non-public information on the investor's judgement. The prediction is the base of the investment stage and the non-public information which can be visited by neighbours. Once the prediction is made, the investor would send orders to exchange, according to its condition of assets (S in equation 5). Both the prediction and investment stage are parts of the investor's state transition function. Prediction stage is more important, because it's the stage of information processing. The heterogeneity of the individuals is also reflected in the prediction stage.

The neighbourhood of the cellular automata is defined as a social network. The initial network is a random graph. However, after each trading day, the network would be rebuilt. An individual's history in-degree and its assets condition ranking are two factors influencing its in-degree in next network. There is a weight number ω to accommodate the importance of the two factors. The pricing mechanism in this model adopts the order-driven Electronic Communications Networks (ECNs) Trading mode.

There are 2 critical control parameters in this artificial financial market, ω and δ_i. ω decides the weight coefficient between historical stickiness and profit orientation when rebuilding the dependency network. δ_i decides the weight coefficient between technical analysis and herd psychology in a cell's prediction stage. The simulation results show that the the co-effect of the two key factors caused the different structures of the neighbourhood, and various features of price fluctuation.

When the individuals' history in-degree plays the main role in rebuilding the neighbourhood network, the degree distribution is shown in Figure 3. After enough trading days, the clustering coefficient is close to 0.5. That means the interaction is active under this condition.

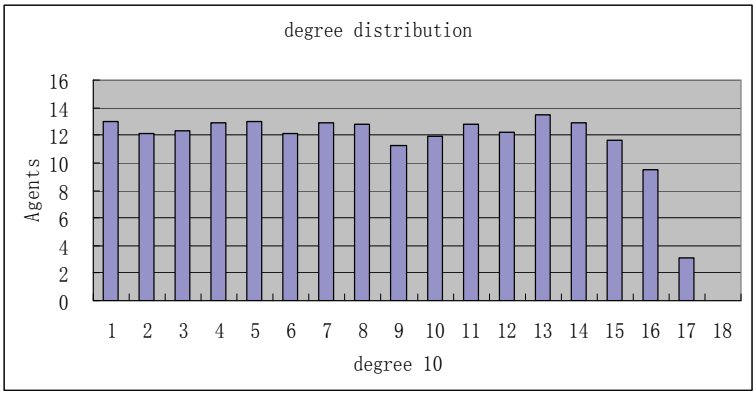


Fig. 3. Degree Distribution of History Degree Ranking Hurt Behaviour

On the other hand, when the individuals' assets condition ranking plays the main role in rebuilding the neighbourhood network, the degree distribution is shown in Figure 4. And the clustering coefficient is close to 0.3. The investors rely more on the judgement of themselves.
Another problem is whether the public or non-public information plays the main role when individuals predict the market price. We simulated the two assumptions both. Combining with the factor of neighbourhood structure, we got four simulation results. We estamited their Hurst exponent using R/S analysis, which are showed in Fig 5, Fig 6, Fig 7, Fig 8.

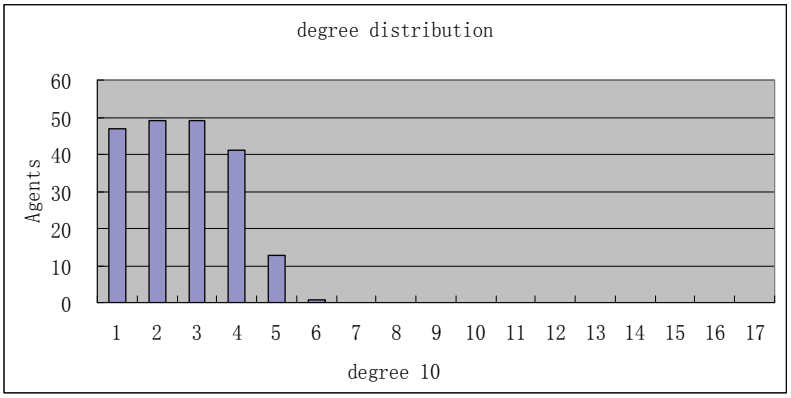


Fig. 4. Degree Distribution of Assets Ranking Hurt Behaviour

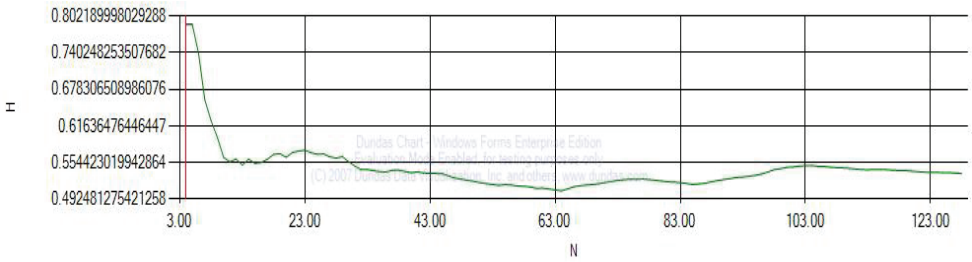


Fig. 5. Assets Ranking Hurt Behaviour & Public information

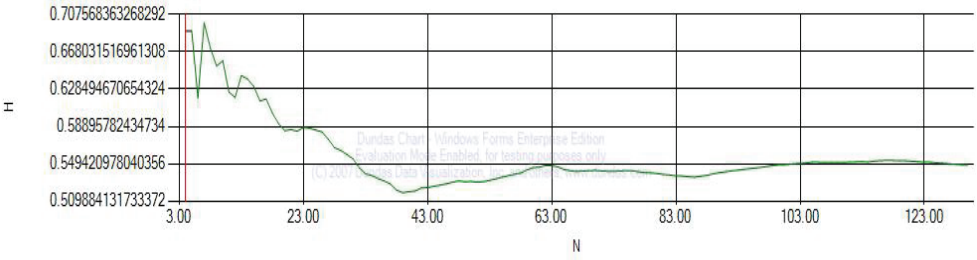


Fig. 6. History Degree Ranking Hurt Behaviour & Public information

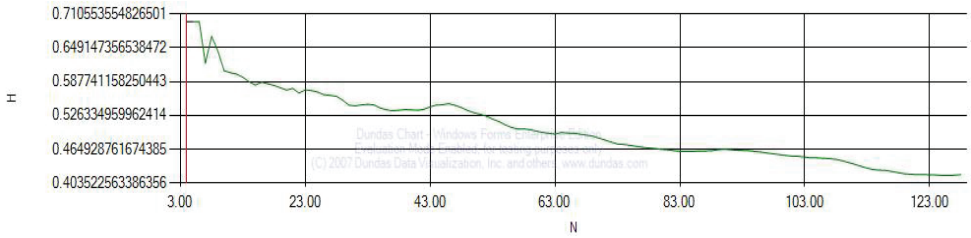


Fig. 7. Assets Ranking Ranking Hurt Behaviour & Non-Public information

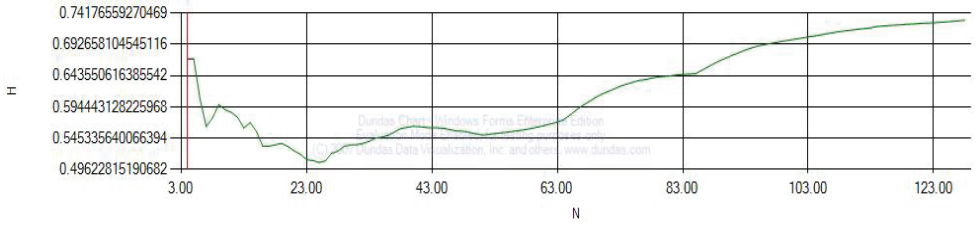


Fig. 8. History Degree Ranking Hurt Behaviour & Non-Public information

It is noticed that the homogeneity of investors is weak when public information plays the primary role in prediction. At this time, especially when assets ranking determines the neighbourhood structure, the Hurst exponent is close to 0.5. This means the volatility of market prices follows Random Walk, just like the hypothesis of classic theory.

When non-public information plays the primary role in prediction., the market mainly consists of herd behavior investors, and the transmission dynamic structure plays the decisive role in determining the durative or anti-durative of the price movement. When assets ranking determine the neighbourhood strucure, the Hurst exponent is close to 0.4 (Figure 7). The remarkable anti-durative of the price movement indicates the collapse of market. When history degree ranking determine the neighbourhood strucure, the historical stickiness causes durative of the price movement, and the Hurst exponent is close to 0.74.

We can also compare the yield rate and the Hurst exponent. As showed in Fig 9, the X-axis is the yield rate and the Y-axis is the Hurst exponent, when Hurst exponent stands low level, yield rate is just a fluctuation around zero. When Hurst exponent is less than 0.5, the system has the feature of anti-persistence, reversals of the price movement would appear frequently.

The simulation result presented above shows that the transmission dynamic structure is of critical importance to the prices movement in a market full of herd behavior investors. Because of the susceptibility of the herd behavior investor, the transmission of the market information could enhance the homogeneity of the investors. If there are some historical sticking authorities trusted by most herd behavior investors in this kind of market, the durative prices movement would appear. If the sticking trust disappears, the herd behavior investor will fall into panic, and the market collapse will come out. It is interesting that both positive and opposite deviation of Hurst exponent from 0.5 is caused by the homogeneity of

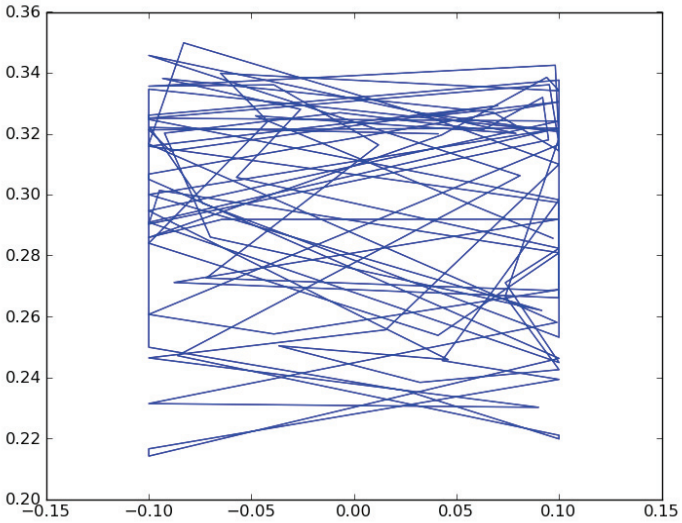


Fig. 9. Phase Diagram of yield rate and the Hurst exponent.

investor structure. The durative or anti-durative of the price movement just depends on whether the information source is keeping changing. In a developed and mature market, however, because there are large amounts of heterogeneous investors, the effect of the transmission dynamic structure is relatively weak.

7. Conclusion and future works

When we build a model for the capital market, tt is difficult to include all essential factors into considering. But we can believe the heterogeneous investors' response to the information spreaded in the market is fundamental motive power of the macrodynamics of the capital market. Investors response to the information and produce information at the same time. The capital market is a kind of self-feedback system. The self-feedback procedure is so complex, and the microstructure formed by the individuals' adaptive behaviour play the primary role. The cellular automata based artificial financial market provided a possibility to describe these factors, and their interaction rules. The self-organization process could be simulated in it. The macrodynamics of the artificial financial market and real capital market can be compared.

However, we must recognize that the cellular automata based artificial market is not mature enough to build a new capital market theory. What is the key feature of the microstructure inside the investors? How can we measure it? There is still no perfect answer. We just have some ideas to do further research. For example, cluster coefficients of network may be related with volatility feature of price; Investors following various behavior rules, may have different average yield rates. But there is still no remarkable result supporting these assumptions.

In the future, the cellular automata based artificial financial market should be extended to describe market factors more particularly. The evolution rule should be more valid. More researches should be focus on the category, form, and spread way of the information. And we should consider more effective way to measure the complexity of the microstructure within the individuals.

8. References

Norman Ehrentreich.(2008) Agent-based modeling: the Santa Fe Institute artificial stock market model revisited, Lecture Notes in Economics and Mathematical Systems, vol 602, Springer Berlin Heidelberg pp.91-112

Yi-ming Wei, Shang-jun Ying, Ying Fan and Bing-Hong Wang.(2003) The cellular automaton model of investment behavior in the stock market, Physica A: Statistical Mechanics and its Applications, Volume 325, Issues 3-4, Pages 507-516

Ying Fan, Shang-Jun Ying, Bing-Hong Wang, Yi-Ming Wei.(2008) The effect of investor psychology on the complexity of stock market: An analysis based on cellular automaton model, Computers & Industrial Engineering.

Edgar E.peters. (1996) Chaos and Order in the Capital Markets: A New View of Cycles, Prices, and Market Volatility, Wiley; 2 edition.

Edgar E.peters.(1994) Fractal Market Analysis: Applying Chaos Theory to Investment and Economics, Wiley; 1 edition.

Jesse Nochella.(2006) Cellular Automata on Networks, NKS 2006 Wolfram Science Conference

M.A. Sánchez Graneroa, J.E. Trinidad Segoviab, and J. García Pérez.(2008) Some comments on Hurst exponent and the long memory processes on capital markets, Physica A: Statistical Mechanics and its Applications, Volume 387, Issue 22, Pages 5543-5551

Daron Acemoglua, Asuman Ozdaglarb, Ali ParandehGheibi. (2010) Spread of (mis)information in social networks, Games and Economic Behavior.

Andrea Consiglio, Annalisa Russino.(June 2007) "How does learning affect market liquidity? A simulation analysis of a double-auction financial market with portfolio traders", Journal of Economic Dynamics and Control, Volume 31, Issue 6, pp. 1910-1937

18

Some Results on Evolving Cellular Automata Applied to the Production Scheduling Problem

Tadeusz Witkowski[1], Arkadiusz Antczak[1],
Paweł Antczak[1] and Soliman Elzway[2]
[1]*Warsaw University of Technology*
[1]*Nasser International University*
[1]*Poland*
[2]*Libya*

1. Introduction

Production scheduling is the process of allocating the resources and then sequencing of task to produce goods. Allocation and sequencing decision are closely related and it is very difficult to model mathematical interaction between them. The allocation problem is solved first and its results are supplied as inputs to the sequencing problem. High quality scheduling improves the delivery performance and lowers the inventory cost. They have much importance in this time based competition. This can be achieved when the scheduling is done in acceptable computation time, but it is difficult because of the NP-hard nature and large size of the scheduling problem.

Based on the machine environment, sequence of operations for the jobs, etc. , the production scheduling problem is divided into the different types: one stage, one process or single machine; one stage, multiple processor or parallel machine; flow shop, job shop, open shop; static and dynamic etc. Job shop is a complex shop where there are finite number of machines, jobs and operation to be done on jobs. There is no direction of flow for jobs. The scheduling is done based on the selection of machine k to process an operation i on job j. Each job can be processed on a machine any number of times. Flexible job-shop scheduling problem (FJSP) extends the JSP by allowing each operations to be processed on more than machine. With this extension, we are now confronted with two subtask: assignment of each operation to an appropriate machine and sequencing operations on each machine.

In the literature, different approaches (tabu search, simulated annealing, variable neighborhood, particle swarm optimization, clonal selection principle etc.) have been proposed to solve this problem (Fattahi,et al., 2007; Kacem, et al., 2002; Liu, et al., 2006; Ong, et. al., 2005; Preissl, 2006; Shi-Jin, et al., 2008; Tay, et al., 2008; Yazdani, et al., 2009). The genetic algorithms (GA), genetic programming, evolution strategies, and evolutionary programming for scheduling problem are described in (Affenzeller, et. al., 2004; Back, et al., 1997; Beham, et al., 2008; Koza, 1992; Mitchell, et. al., 2005; Zomaya, et. al., 2005; Stocher, et. al., 2007; Winkler, et. al., 2009), and cellular automata are presented in (De Castro, 2006; Tomassini, 2000; Seredyński, 2002). Using GA algorithm to behavior in cellular automata (CA), evolutionary design of rule changing CA, and other problems are described in (Back,

et. al., 2005; Kanoh, et. al., 2003; Martins, et. al., 2005; Das, et. al., 1994; Sipper, 1997,1999; Subrata, et. al., 2003; Sahoo, et. al. 2007).
The difficulty of designing cellular automatons transition rules to perform a particular problem has severely limited their applications.
In (Seredyński,et. al., 2002) evolution of cellular automata-based multiprocessor scheduling algorithm is created. In learning mode a GA is applied to discover rules of CA suitable for solving instances of a scheduling problem. In operation mode discovered rules of CA are able to find automatically an optimal or suboptimal solution of the scheduling problem for any initial allocation of a program graph in two-processor system graph.
The evolutionary design of CA rules has been studied by th EVCA group in detail. A genetic algorithm GA was used to evolve CAs for the two computational tasks. The GA was shown to have discovered rules that gave rise to sophisticated emergent computational strategies. Sipper (1999) has studied a cellular programming algorithm for 2-state non-uniform CAs, in which each cell may contain a different rule. The evolution of rules is here performed by applying crossover and mutation. He showed that this method is better than uniform (ordinary) Cas with a standard GA for the two tasks. In Kanoh (2003) was proposed a new programming method of cellular computers using genetic algorithms. Authors considered a pair of rules and the number of rule iterations as a step in the computer program. This method im meant to reduce the complexity of a given problem by dividing the problem into smaller ones and assigning a distinct rule to each.
This study introduces an approach to solving evolutionary cellular automata-based FJSP. In this paper genetic programming is applied in this algorithm – rule tables undergo selection and crossover operations in the populations that follow.
The paper is organized as follows. Section 2 gives formulation of the problem. A formal definition of CA is described in section 3. Section 4 explains the details of the evolving CA-based production scheduling. Section 5 shows the computational results and the comparison of CA and GA for finding solutions in FJSP is presented. Some concluding remarks are given in section 6.

2. Problem formulation

The FJSP is formulated as follows. There is a set of jobs $Z = \{Z_i\}$, $i \in I$, where $I = \{1, 2, ..., n\}$ is an admissible set of parts, $U = \{u_k\}$, $k \in 1, m$, is a set of machines. Each job Z_i is a group of parts Π_i of equal partial task p_i of a certain range of production. Operations of technological processing of the i-th part are denoted by $\{O_{ij}\}_{j=\xi}^{H_i}$. Then for Z_i , we can write $Z_i = (\Pi_i \{O_{ij}\}_{j=\xi}^{H_i})$, where $O_{ij} = (G_{ij}, t_{ij}(N))$ is the j-th operation of processing the i-th group of parts; ξ_i is the number of operation of the production process at which one should start the processing the i- th group of parts; H_i is the number of the last operation for a given group; G_{ij} is a group of interchangeable machines that is assigned to the operation O_{ij}; G is a set of all groups of machines arose in the matrix $||\{Z_i\}||$; $t_{ij}(N)$ is an elementary duration of the operation O_{ij} with one part d_i that depends on the number of machine N in the group (on the specified operations); t'_{ij} is the duration of set up before the operation O_{ij}; N_{gr} is the number of all groups of machines. The most widely used objective is to find feasible schedules that minimize the completion time of the total production program, normally referred to as makespan (C_{max}).

3. Formal definition cellular automata

A d-dimensional CA consists of a finite or infinite d-dimensional grid of cells, each of which can take on a value from a finite, usually small, set of integers. The value of each cell at time step $t + 1$ is a function of the values of small local neighborhood of cells at time t. The cells update their state simultaneously according to a given local rule. Formally, a CA can be defined as a quintuple (De Castro, 2006)

$$C = <S, s_0, G, d, f>$$

where S is a finite set of states, $s_0 \in S$ are the initial states of the CA, G is cellular neighborhood, $d \in Z+$ is the dimension of C, and f is the local cellular interaction rule, also referred to as the *transition function* or *transition rule*. Given the position of a cell **i**, where **i** is an integer vector in a d-dimensional space ($\mathbf{i} \in Zd$), in a regular d-dimensional uniform lattice, or grid, its neighborhood G is defined by

$$G_{\mathbf{i}} = \{\mathbf{i}, \mathbf{i}+ \mathbf{r}_1, \mathbf{i}+ \mathbf{r}_2,..., \mathbf{i}+ \mathbf{r}_n\}$$

where n is a fixed parameter that determines the neighborhood size, and $\mathbf{r}_j$ is a fixed vector in the d-dimensional space. The local transition rule f

$$f : Sn \rightarrow S$$

maps the state $s_{\mathbf{i}} \in S$ of a given cell **i** into another state from the set S, as a function of the states of the cells in the neighborhood $G_{\mathbf{i}}$. In a uniform CA, f is identical for all cells, whereas in nonuniform CA, f may differ from one cell to another, i.e., f depends on **i**, $f_{\mathbf{i}}$. For a finite-size CA of size N, where N is the number of cells in the CA, a configuration of the grid at time t is defined as

$$C(t) = (s_0(t), s_1(t),...,s_{N-1}(t))$$

where $s_{\mathbf{i}}(t)$ is the state of cell **i** at time t**.** The progression of the CA in time is then given by the iteration of the global mapping F

$$F : C(t) \rightarrow C(t+1), \quad t = 0,1,...$$

Through the simultaneous application in each cell of the local transition rule f, the global dynamics of the CA can be described as a directed graph, referred to as the CA' s state space. One-and bi-dimensional CA are the most usually explored types of CA. In the one-dimensional case, there are usually only two possible states for each cell, $S = \{0,1\}$. Thus, f is a function $f : \{0,1\}$ n $\rightarrow \{0,1\}$ and the neighborhood size n is usually taken to be $n = 2r+1$ such that

$$s_{\mathbf{i}}(t+ 1) = (s_{\mathbf{i}-r}(t),...,s_{\mathbf{i}}(t),...,s_{\mathbf{i}+r}(t))$$

where $r \in Z+$ is a parameter, known as the radius, representing the standard one-dimensional cellular neighborhood.

4. Evolving cellular automata for FJSP

4.1 Algorithm for evolving CA for FJSP

The general working principle of evolutionary algorithms is based on a program loop that involves implementations of the operators mutation, recombination, selection, and fitness evaluation on a set of candidate solutions for a given problem.

The algorithm which generates the schedule bases on two CAs. One is responsible for construction sequencing operations on individual parts, and the other for the allocation of machines to operation with interchangeably group machines.

The crossover operation is realized on the current and previous population using a definite number of the best rules in the two above-mentioned populations. Half of that definite number is taken from the current population, and the other half from the previous one.

Depending on the generated value and the determined intensity the re-writing of the values from the current table to the previous one or vice versa takes place (no operation is also possible). During the algorithm operation in a loop state changes of the CA are executed basing on the transition tables.

They define the change of the current position of an element in the state table on the basis of its current value. The repetition of the operation causes changes in the CA state, which defines the sequence of technological operations and machines used. On the basis of those state tables a proper schedule is generated (reservation of machines).

Genetic algorithm is applied in the CA algorithm – rule tables undergo selection and crossover operations in the populations that follow.

The algorithm sequences the technological operations on a given set of parts of different kinds using evolving CAs. This is realized with the use a genetic algorithm which performs a selection of the so-called transition tables (i.e. rule tables, state change tables) of the two cellular automata whose functions are described above.

The input parameters are: the number of the population of automata transition tables (rule tables - RT), the number of populations, the number of transitions, the hybrid coefficient (the number of the tables in the populations being crossed over with a given probability), the hybridization intensity (the probability of the crossover operation on given elements of the tables). Fig. 1 shows the flowchart of evolving CA used to create schedules.

The algorithm is based on two cellular automata: a) determination of machine allocation from the interchangeable group for individual operations, b) determination of part sequence for individual operations.

The CA state change is realized as follows. Let o_{i1},o_{i2} position in the state table (ST) where o_{i1}=0. We determine the value n = Operation $_{i}$ [o_{i1}], where n = Operation $_{i\,\mathrm{ST}}$ [o_{i1}], which we use to calculate D where D = Operation $_{i\,\mathrm{RT}}$ [n]. We calculate this position o_{i2} from the formula o_{i2} = (o_{i1} +D) mod N (where N – number of jobs). Next the value change in the CA state table is realized on the above mentioned positions o_{i1}, o_{i2}:

$$\text{Operation}_{i\,\mathrm{ST}}[o_{i1}] = \text{Operation}_{i\,\mathrm{ST}}[o_{i2}];$$

eg. for o_{i1} = o_{i2} for the values 0 1 2 3 4 5 6 in the previous state table we have values 5 1 2 3 4 0 6 in the new state table. After the change is done we assume o_{i1} = o_{i2}. All the last permutations obtained as a result of the CA execution for each of the operations of all the CA state tables create a schedule. The number of schedules in one iteration of the algorithm is equal to number of populations. At the next stage schedule sorting takes place on the basis of the value of the makespan as well as the their selection with a determined

hybridization coefficient. As a result of those operations new rule tables for the next iteration are obtained. The CA for machine choice in individual groups operates in a similar way.

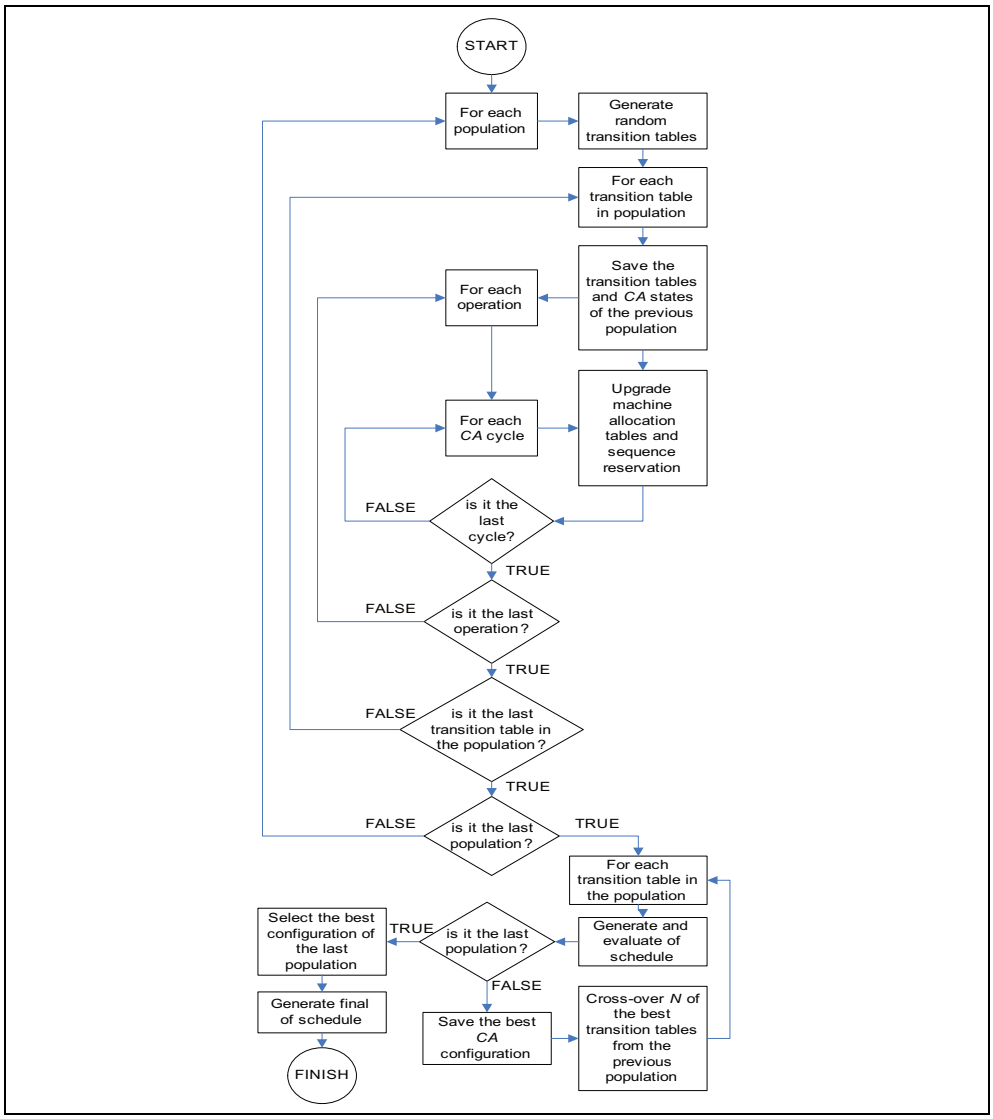


Fig. 1. Flowchart of evolving CA for flexible job shop scheduling.

4.2 Example

Examples of transition tables for the CA responsible for machine allocation from technological groups for individual operations are shown in Fig. 2.

Transition table: 0
Operation: 0 [2, 0, 5, 0, 4, 4, 2]
Operation: 1 [1, 1, 4, 5, 5, 2, 1]
Operation: 2 [5, 1, 5, 2, 6, 1, 5]
Operation: 3 [5, 2, 6, 0, 2, 2, 6]
Transition table: 1
Operation: 0 [2, 1, 6, 2, 0, 0, 3]
Operation: 1 [0, 2, 2, 5, 5, 4, 5]
Operation: 2 [1, 1, 0, 4, 0, 6, 3]
Operation: 3 [4, 3, 2, 3, 1, 3, 0]
Transition table: 2
Operation: 0 [4, 1, 2, 3, 0, 6, 1]
Operation: 1 [0, 0, 0, 2, 6, 0, 3]
Operation: 2 [6, 1, 2, 3, 5, 2, 0]
Operation: 3 [6, 5, 4, 3, 3, 6, 0]

Fig. 2. Automata transition tables (allocate machines)

Examples of transition tables for the CA responsible for operation sequence in a generated schedule are shown in Fig. 3.

Transition table: 0
Operation: 0 [5, 4, 1, 0, 1, 2, 6]
Operation: 1 [3, 2, 2, 4, 0, 6, 5]
Operation: 2 [6, 2, 1, 5, 0, 3, 1]
Operation: 3 [4, 6, 2, 5, 2, 0, 6]
Transition table: 1
Operation: 0 [6, 5, 3, 0, 3, 2, 2]
Operation: 1 [6, 4, 0, 1, 5, 0, 3]
Operation: 2 [4, 1, 5, 3, 3, 4, 5]
Operation: 3 [6, 2, 6, 3, 6, 6, 4]
Transition table: 2
Operation: 0 [5, 2, 4, 6, 3, 2, 3]
Operation: 1 [2, 2, 6, 3, 2, 0, 4]
Operation: 2 [0, 2, 4, 2, 0, 3, 1]
Operation: 3 [5, 5, 6, 6, 3, 3, 5]

Fig. 3. Automata transition tables (sequence operations)

The use of machines (CA changes for the 10 consecutive states in a cycle of each table - left column) and the operation sequence (CA changes for the next consecutive states in a cycle of each table -right column) for one population are shown in Fig.4.

State table: 0

Op: 0 [0, 0, -1, 0, -1, -1, -1]	**Op: 0 [0, 1, 2, 3, 4, 5, 6]**
Op: 0 [1, 0, -1, 0, -1, -1, -1]	Op: 0 [5, 1, 2, 3, 4, 0, 6]
Op: 0 [1, 0, -1, 0, -1, -1, -1]	Op: 0 [5, 1, 2, 0, 4, 3, 6]
.....	.....
Op: 0 [1, 0, -1, 1, -1, -1, -1]	Op: 0 [3, 6, 5, 0, 2, 1, 4]
Op: 0 [1, 0, -1, 2, -1, -1, -1]	Op: 0 [3, 0, 5, 6, 2, 1, 4]

Op: 1 [0, 0, 0, -1, 0, -1, -1]	**Op: 1 [0, 1, 2, 3, 4, 5, 6]**
Op: 1 [1, 0, 0, -1, 0, -1, -1]	O p: 1 [2, 1, 0, 3, 4, 5, 6]
Op: 1 [1, 0, 0, -1, 0, -1, -1]	Op: 1 [2, 1, 4, 3, 0, 5, 6]
.....	.....
Op: 1 [0, 0, 0, -1, 0, -1, -1]	Op: 1 [4, 3, 6, 5, 0, 2, 1]
Op: 1 [1, 0, 0, -1, 0, -1, -1]	Op: 1 [4, 3, 6, 5, 1, 2, 0]

Op: 2 [-1, -1, -1, 0, 0, 0, 0]	**Op: 2 [0, 1, 2, 3, 4, 5, 6]**
Op: 2 [-1, -1, -1, 0, 0, 0, 0]	Op: 2 [6, 1, 2, 3, 4, 5, 0]
Op: 2 [-1, -1, -1, 0, 0, 0, 0]	Op: 2 [6, 1, 2, 3, 4, 0, 5]
.....	.....
Op: 2 [-1, -1, -1, 2, 1, 2, 1]	Op: 2 [5, 6, 1, 2, 3, 0, 4]
Op: 2 [-1, -1, -1, 2, 1, 2, 2]	Op: 2 [5, 6, 1, 2, 0, 3, 4]

Op: 3 [-1, -1, -1, -1, 0, 0, 0]	**Op: 3 [0, 1, 2, 3, 4, 5, 6]**
Op: 3 [-1, -1, -1, -1, 0, 0, 0]	Op: 3 [4, 1, 2, 3, 0, 5, 6]
Op: 3 [-1, -1, -1, -1, 0, 0, 0]	Op: 3 [4, 0, 2, 3, 1, 5, 6]
.....	.....
Op: 3 [-1, -1, -1, -1, 1, 1, 1]	Op: 3 [1, 0, 6, 4, 5, 2, 3]
Op: 3 [-1, -1, -1, -1, 1, 1, 1]	Op: 3 [1, 2, 6, 4, 5, 0, 3]

State table: 1

Op: 0 [0, 0, -1, 0, -1, -1, -1]	**Op: 0 [0, 1, 2, 3, 4, 5, 6]**
Op: 0 [1, 0, -1, 0, -1, -1, -1]	Op: 0 [6, 1, 2, 3, 4, 5, 0]
Op: 0 [1, 0, -1, 0, -1, -1, -1]	Op: 0 [6, 1, 2, 3, 4, 0, 5]
.....	.....
Op: 0 [2, 0, -1, 2, -1, -1, -1]	Op: 0 [5, 6, 1, 2, 3, 0, 4]
Op: 0 [2, 0, -1, 2, -1, -1, -1]	Op: 0 [5, 6, 1, 2, 0, 3, 4]

Op: 1 [0, 0, 0, -1, 0, -1, -1]	**Op: 1 [0, 1, 2, 3, 4, 5, 6]**
Op: 1 [1, 0, 0, -1, 0, -1, -1]	Op: 1 [2, 1, 0, 3, 4, 5, 6]
Op: 1 [1, 0, 0, -1, 0, -1, -1]	Op: 1 [2, 1, 4, 3, 0, 5, 6]
.....	.....
Op: 1 [2, 0, 0, -1, 0, -1, -1]	Op: 1 [4, 3, 6, 5, 0, 2, 1]
Op: 1 [2, 0, 0, -1, 0, -1, -1]	Op: 1 [4, 3, 6, 5, 1, 2, 0]

Op: 2 [-1, -1, -1, 0, 0, 0, 0]	**Op: 2 [0, 1, 2, 3, 4, 5, 6]**
Op: 2 [-1, -1, -1, 0, 0, 0, 0]	Op: 2 [4, 1, 2, 3, 0, 5, 6]
Op: 2 [-1, -1, -1, 0, 0, 0, 0]	Op: 2 [4, 0, 2, 3, 1, 5, 6]

.....	.....
Op: 2 [-1, -1, -1, 2, 0, 0, 0]	Op: 2 [1, 0, 6, 4, 5, 2, 3]
Op: 2 [-1, -1, -1, 2, 0, 0, 0]	Op: 2 [1, 2, 6, 4, 5, 0, 3]
Op: 3 [-1, -1, -1, -1, 0, 0, 0]	**Op: 3 [0, 1, 2, 3, 4, 5, 6]**
Op: 3 [-1, -1, -1, -1, 0, 0, 0]	Op: 3 [6, 1, 2, 3, 4, 5, 0]
Op: 3 [-1, -1, -1, -1, 0, 0, 0]	Op: 3 [6, 1, 2, 3, 4, 0, 5]
.....	.....
Op: 3 [-1, -1, -1, -1, 1, 1, 0]	Op: 3 [5, 6, 1, 2, 3, 0, 4]
Op: 3 [-1, -1, -1, -1, 2, 1, 0]	Op: 3 [5, 6, 1, 2, 0, 3, 4]

State table: 2

Op: 0 [0, 0, -1, 0, -1, -1, -1]	**Op: 0 [0, 1, 2, 3, 4, 5, 6]**
Op: 0 [1, 0, -1, 0, -1, -1, -1]	Op: 0 [5, 1, 2, 3, 4, 0, 6]
Op: 0 [1, 0, -1, 0, -1, -1, -1]	Op: 0 [5, 1, 2, 0, 4, 3, 6]
.....	.....
Op: 0 [0, 0, -1, 0, -1, -1, -1]	Op: 0 [3, 6, 5, 0, 2, 1, 4]
Op: 0 [0, 0, -1, 0, -1, -1, -1]	Op: 0 [3, 0, 5, 6, 2, 1, 4]
Op: 1 [0, 0, 0, -1, 0, -1, -1]	**Op: 1 [0, 1, 2, 3, 4, 5, 6]**
Op: 1 [1, 0, 0, -1, 0, -1, -1]	Op: 1 [4, 1, 2, 3, 0, 5, 6]
Op: 1 [1, 0, 0, -1, 0, -1, -1]	Op: 1 [4, 0, 2, 3, 1, 5, 6]
.....	.....
Op: 1 [0, 0, 0, -1, 0, -1, -1]	Op: 1 [1, 0, 6, 4, 5, 2, 3]
Op: 1 [1, 0, 0, -1, 0, -1, -1]	Op: 1 [1, 2, 6, 4, 5, 0, 3]
Op: 2 [-1, -1, -1, 0, 0, 0, 0]	**Op: 2 [0, 1, 2, 3, 4, 5, 6]**
Op: 2 [-1, -1, -1, 0, 0, 0, 0]	Op: 2 [0, 1, 2, 3, 4, 5, 6]
Op: 2 [-1, -1, -1, 0, 0, 0, 0]	Op: 2 [0, 1, 2, 3, 4, 5, 6]
.....	.....
Op: 2 [-1, -1, -1, 1, 0, 0, 2]	Op: 2 [0, 1, 2, 3, 4, 5, 6]
Op: 2 [-1, -1, -1, 1, 0, 0, 0]	Op: 2 [0, 1, 2, 3, 4, 5, 6]
Op: 3 [-1, -1, -1, -1, 0, 0, 0]	**Op: 3 [0, 1, 2, 3, 4, 5, 6]**
Op: 3 [-1, -1, -1, -1, 0, 0, 0]	Op: 3 [5, 1, 2, 3, 4, 0, 6]
Op: 3 [-1, -1, -1, -1, 0, 0, 0]	Op: 3 [5, 1, 2, 0, 4, 3, 6]
.....	.....
Op: 3 [-1, -1, -1, -1, 1, 0, 0]	Op: 3 [3, 6, 5, 0, 2, 1, 4]
Op: 3 [-1, -1, -1, -1, 2, 0, 0]	Op: 3 [3, 0, 5, 6, 2, 1, 4]

Fig. 4. CA changes for the 10 consecutive states

The numbers in the left column of the tables stand for the number of the machine in a group, and their indexes (i.e. allocation in the table) are the numbers of the parts. The numbers in the right column of the tables stand for the sequence of individual parts in a given operation and their indexes (i.e. allocation in the table) are the numbers of the parts. Value (-1) in the left column of the tables stands for lack of machine participation in a given operation with a given part. Value (-1) in the right column would stand for lack of processing of a part in a given operation. All the (-1) values are ignored in the state change

procedure of the CA, and does not participate in the machine allocation procedure. For each iteration makespan is determined for the generated schedule, on the basis of the final states of both automata.

All the makespan**es** for each schedule from a population are recorded and compared in order to select the best schedule from the current population. If it is not the final population then the best rule tables are crossed over in order to generate the best schedules from the current and previous population. In each iteration summary time realize of all operations (makespan) for generate schedule on basis final state two cellular automata is determinated. All makespan**es** for each schedule with population are writing and compare to aim choice best schedule among current populations. If population no is latest we realize crossower operation best rule table which lead for generated best schedules with current and previous populations. Half given number is taken with current population, and second half with previous population. Depending to generated value and given intensity follows determine values with current table to previous table or vice versa (is possible lack operation).

5. Computational results

5.1 Comparative study of cellular automata for FJSP

Two types of routing were considered: a serial and a parallel one. In a serial route an entire batch of parts is processed on one machine and only when all of the products in the batch have been processed are they sent to the next machine. In a parallel route individual items of the batch are sent to the next machines as soon as they have been processed on the previous machine.

The research was carried out on a computer with an Intel Core2 2.4 GHz processor and 2047 MB of RAM for the following settings of the CA algorithm: size of population = 1000; number of iterations = 100; number of transitions = 1000; hybridization ratio = 0.9; and intensity of hybridization = 0.9.

For solution of FJSP problem special software to realize the CA algorithm have been created. Computer experiments were carried out for data presented in (Witkowski, 2005) – where the number of operations is 160, and the number of machines 26.

The experiments have been carried out for the hybridization ratio: 0,1; 0,5; 0,9 and the intensity of hybridization equal to 0,1; 0,5; 0,9. The simulation of each test problem was run with the SP population size equal to 10, 100, 1000, the RT transition rate was equal to 10, 100, 1000, and the IN iteration number was equal to 10, 100, 1000. Besides, in some cases the values of SP, RT and IT reached 10000.

The following symbols for signed algorithm parameters: SP - size of population; IT - number of iterations; RT - number of transitions; HR - hybridization ratio; and IH - intensity of hybridization have been used. Individual SP, IT and RT parameters assume one of the values from the set {10, 100, 1000}; moreover HR and IH from the sets{(0,1; 0,1); (0,5; 0,5) and (0,9; 0,9)} respectively.

For IT, SP and RT values we assume the following linguistic variables : N – low value, S – medium value, D – high value (V – very high value – in some combinations of parameters). In this way 27 combinations with parameters of the algorithm were created (fig. 5) For example. one of such combinations is IT(M)-SP(S)-RT(D), etc..

Table 1 shows some of the results (with the SP (D) value) for the test parameters of the CA algorithm.

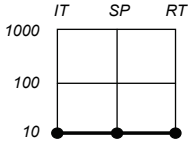
hybridization ratio = 0.9; intensity of hybridization = 0.9. size of the population = 1000; number of transitions = 10.									
iter.	*Makespan*					*min.*	*average*	*max.*	*avg. time [sec.]*
10	66295.6	71146.1	71852.0	69426.5	69729.6	66295.6	69282.3	71852.0	10
	69158.2	68452.7	69949.2	66517.3	70295.7				
100	67436.6	63232.1	65860.1	69623.8	63542.7	63232.1	66403.6	69623.8	102
	66320.0	68723.9	65948.9	66832.0	66516.2				
1000	66305.5	66918.2	64587.4	65644.4	63057.3	63057.3	66176.2	68121.8	1020
	67373.0	68121.8	66553.8	67074.6	66126.4				

hybridization ratio = 0.9; intensity of hybridization = 0.9. size of the population = 1000; number of transitions = 100.									
iter.	*Makespan*					*min.*	*average*	*max.*	*avg. time [sec.]*
10	72254,0	68692,3	72917,7	69880,7	69518,9	66240,2	69449,3	72917,7	10
	71397,2	66240,2	67804,5	68055,5	67732,4				
100	70885,2	68269,0	63363,8	65451,6	70035,0	63363,8	67352,8	70885,2	106
	67585,8	67766,4	66859,0	68564,2	64747,8				
1000	66239,3	68417,1	69373,5	68334,9	62826,0	62826,0	67515,4	69373,5	1068
	69118,0	69067,4	68188,7	67135,2	66453,5				

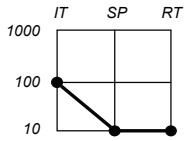
hybridization ratio = 0.9; intensity of hybridization = 0.9. size of the population = 1000; number of transitions = 1000.									
iter.	*Makespan*					*min.*	*average*	*max.*	*avg. time [sec.]*
10	70217.3	66442.4	68170.6	69813.4	65720.8	63161.2	68811.1	72895.0	15
	63161.2	71086.2	72199.9	68404.2	72895.0				
100	67537.3	62753.4	71358.4	67292.3	67810.3	62753.4	67242.0	71358.4	156
	69405.4	66784.8	65753.1	68183.8	65540.8				
1000	64347.2	63539.3	66144.4	66560.0	69469.2	63539.3	67083.4	70579.7	1564
	67208.2	70579.7	66920.0	67386.3	68679.3				

Table 1. Some of the results (with the SP (D) value) for the test parameters of the CA algorithm (serial route)

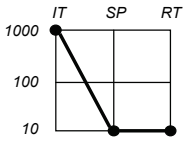
Figure 5 summarizes the results for the test problems that were run with the evolving cellular automata algorithm for the serial route.



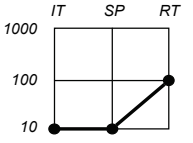
0.1; 0.1	81662,7
0.5; 0.5	81044,9
0.9; 0.9	78554,2



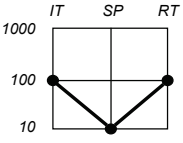
0.1; 0.1	77798,7
0.5; 0.5	76117,5
0.9; 0.9	78357,8



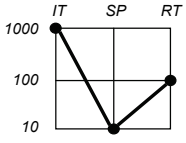
0.1; 0.1	76975,0
0.5; 0.5	79498,8
0.9; 0.9	77680,8



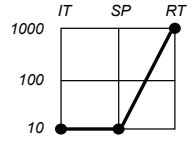
0.1; 0.1	83589,0
0.5; 0.5	83247,5
0.9; 0.9	81261,1



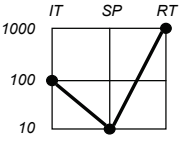
0.1; 0.1	79370,1
0.5; 0.5	81223,4
0.9; 0.9	80910,9



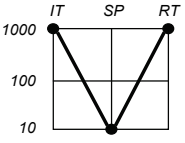
0.1; 0.1	80219,5
0.5; 0.5	83431,1
0.9; 0.9	77384,0



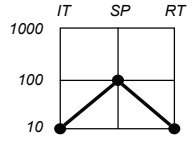
0.1; 0.1	82118,8
0.5; 0.5	79335,4
0.9; 0.9	83613,9



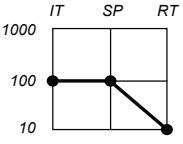
0.1; 0.1	81935,2
0.5; 0.5	78669,3
0.9; 0.9	78325,0



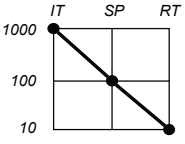
0.1; 0.1	78687,1
0.5; 0.5	82720,1
0.9; 0.9	80292,7



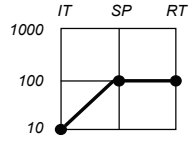
0.1; 0.1	76069,6
0.5; 0.5	75395,4
0.9; 0.9	74073,0



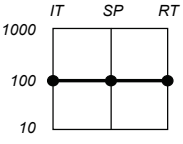
0.1; 0.1	69121,4
0.5; 0.5	70828,1
0.9; 0.9	70462,8



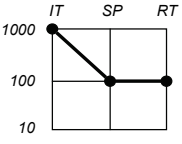
0.1; 0.1	71081,5
0.5; 0.5	70800,4
0.9; 0.9	70602,6



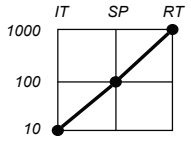
0.1; 0.1	74129,5
0.5; 0.5	76410,9
0.9; 0.9	74623,5



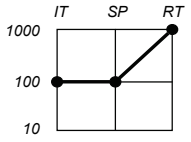
0.1; 0.1	70526,7
0.5; 0.5	73550,7
0.9; 0.9	73124,9



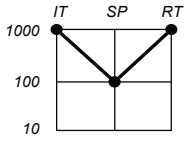
0.1; 0.1	72133,1
0.5; 0.5	72126,3
0.9; 0.9	73429,8



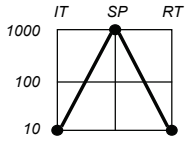
0.1; 0.1	75249,7
0.5; 0.5	73438,5
0.9; 0.9	75201,3



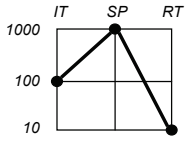
0.1; 0.1	72750,8
0.5; 0.5	72288,4
0.9; 0.9	71287,6



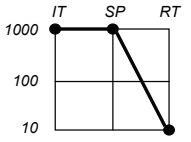
0.1; 0.1	73102,2
0.5; 0.5	73647,3
0.9; 0.9	72543,6



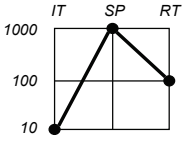
0.1; 0.1	69027,6
0.5; 0.5	69151,0
0.9; 0.9	69282,3



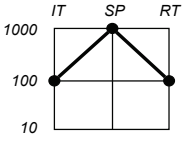
0.1; 0.1	67350,0
0.5; 0.5	66859,7
0.9; 0.9	66403,6



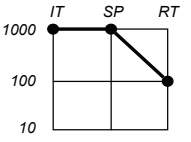
0.1; 0.1	66282,5
0.5; 0.5	66526,8
0.9; 0.9	66176,2



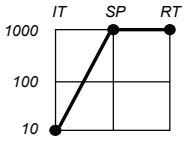
0.1; 0.1	69748,7
0.5; 0.5	69836,3
0.9; 0.9	69449,3



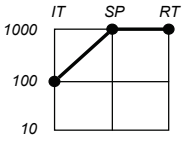
0.1; 0.1	68200,7
0.5; 0.5	67070,3
0.9; 0.9	67352,8



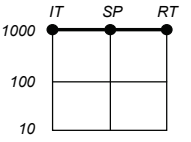
0.1; 0.1	66575,2
0.5; 0.5	66756,9
0.9; 0.9	67515,4



0.1; 0.1	70648,9
0.5; 0.5	69980,9
0.9; 0.9	68811,1



0.1; 0.1	67888,8
0.5; 0.5	68057,5
0.9; 0.9	67242,0



0.1; 0.1	67882,8
0.5; 0.5	67532,3
0.9; 0.9	67083,4

Fig. 5. The results for the test problems that were run with the evolving cellular automata algorithm (serial route)

We can generally see that depending on the PS population size we can single out 3 classes of quality results (with regard to the C_{max} criterion) - very good (large population size), average (medium population size) and poor (small population size). Moreover an increase of the IT value influences the C_{max} more than the RT value, although there are a number of exceptions.
The best results of C_{max} are always obtained at SP(D) regardless of the IT or RT values. Eg. at SP(D) value the best C_{max} is achieved for combination IT(D)-SP(D)-RT(M) rather than for IT(M)-SP(D)-RT(D); at SP(D) value the best C_{max} is achieved for combination IT(S)-SP(D)-RT(M) rather than for IT(M)-SP(D)-RT(S); at SP(D) value the best C_{max} is achieved for combination IT(D)-SP(D)-RT(S) rather than for IT(S)-SP(D)-RT(D). It should be noted that

for combination IT(D)-SP(D)-RT(D) an insignificantly poor C_{max} is achieved than for eg. IT(D)-SP(D)-RT(M).

The worst results of C_{max} are always obtained at SP(M) regardless of the IT or RT values. Eg. at SP(M) value the best C_{max} is achieved for combination IT(D)-SP(M)-RT(M) rather than for IT(M)-SP(M)-RT(D); moreover for combination IT(S)-SP(M)-RT(D) the C_{max} values are better than for IT(D)-SP(M)-RT(S) – while the pair (HR,IH) = (0,5:0,5), and the C_{max} is worse while the pair (HR,IH) = (0,1;0,1). For combination IT(S)-SP(M)-RT(M) the C_{max} values are clearly better than for IT(M)-SP(M)-RT(S). We can also see that for IT(D)-SP(D)-RT(D) an insignificantly poor C_{max} is achieved than eg. for IT(D)-SP(D)-RT(M).

Analizing the influence of SP on the C_{max} we can observe the following behaviour of the CA algorithm. An increase of the SP value from 10 to 100 decreases the average value of C_{max} from ca. 82000 to 74000 min., i.e. by about 8000 min. An increase of the SP value from 100 to 1000 decreases the average C_{max} value from 74000 to 69000 min. - by about 5000 min. Thus an increase of the SP value from 10 to 1000 decreases the average C_{max} value from 82000 to 69000 min. i.e. by about 13000 min. We can see that the increase from 100 to 1000 results in a slower decrease of C_{max} (i.e. by about 5000 min.) than the change of the SP value from 10 to 100 (i.e. about 8000 min.).

Let us consider the influence of IT on the C_{max} value. For combinations with SP(M) and RT(M) an increase of IT from 10 to 100 results in a decrease of C_{max} from 80000 to 77000 min., i.e. by ca. 3000 min. An IT increase from 100 to 1000 results in an insignificant decrease of C_{max} - by about 500 min. - and while the pair (HR,IH) = (0.5;0,5) in an increase of C_{max}. For combination with SP(S) and RT(M) the change of IT from z 10 to 100 gives an decrease of C_{max} from 75000 to 70000 min., i.e by ca. 5000 min.; moreover an increse of the IT value from 100 to 1 000 gives an insignificant decrease of C_{max} while (HR,IH) = (0.5;0,5) and a decrease of C_{max} while (HR,IH) = (0,1;01) and (HR,IH) = (0,9;0,9). At SP(D) and RT(M) values the increase of IT from 10 to 100 gives a decrease of C_{max} from 69000 to 67000 min., ie. by ca. 2000 min. An increase of IT from 100 to 1000 decreases the C_{max} from 67000 to 66500 min. for combinations IT(M)-SP(D)-RT(M), IT(S)-SP(D)-RT(M) and IT(D)-SP(D)-RT(M). A similar situation occurs for combinations IT(M)-SP(D)-RT(D), IT(S)-SP(D)-RT(D) and IT(D)-SP(D)-RT(D).

An increase of the IT value in most cases improves the C_{max} value eg. for combinations at SP(D), but there are also exceptions. For combination IT(S)-SP(D)-RT(S) we have better C_{max} than for IT(M)-SP(D)-RT(S). Moreover the C_{max} value increases in the following order: from IT(D)-SP(D)-RT(M) to IT(S)-SP(D)-RT(M) to IT(M)-SP(D)-RT(M). An increase of the IT value does not always result in a better C_{max}. For example dla C_{max} with average values i.e. with combinations which have the medium parameter SP(S) combination IT(S)-SP(S)-RT(S) gives a better C_{max} than IT(D)-SP(S)-RT(S) and combination IT(S)-SP(S)-RT(D) gives a better C_{max} than IT(D)-SP(S)-RT(D). Moreover combination IT(S)-SP(S)-RT(M) gives a better C_{max} than IT(D)-SP(S)-RT(M) while the pair (HR,IH) = (0,1;0,1) and the pair (HR,IH) = (0,9;0,9).

The increase of the RT value both increases and decreases the C_{max} value. For example combination IT(M)-SP(S)-RT(D) gives a better C_{max} than IT(M)-SP(S)-RT(S) while the pair (HR,IH) = (0,5;0,5) and IT(S)-SP(S)-RT(D) gives a better C_{max} than IT(S)-SP(S)-RT(M) while the pair (HR,IH) = (0,5;0,5) and (HR, IH) = (0,9;0,9). We can also note the following cases: combination IT(D)-SP(S)-RT(D) gives better values of C_{max} than IT(D)-SP(S)-RT(S); IT(D)-SP(M)-RT(S) gives better C_{max} than IT(D)-SP(M)-RT(M) while the pair (HR,IH) = (0,9;0,9); IT(D)-SP(M)-RT(D) gives a better C_{max} than IT(D)-SP(M)-RT(S) while the pair (HR,IH) = (0,5;0,5) and (HR,IH) = (0,9;0,9); IT(S)-SP(M)-RT(D) gives a better C_{max} than IT(S)-SP(M)-

RT(S) while the pair (HR,IH) = (0,5;0,5) and (HR,IH) = (0,9;0,9); IT(M)-SP(M)-RT(D) gives a better C_{max} than IT(M)-SP(M)-RT(M) while (HR, IH) = (0,1;0,1) and (HR, IH) = (0,9;0,9) and a better C_{max} than IT(M)-SP(M)-RT(M) while (HR,IH) = (0,5;0,5).
For IT(D)-SP(D)-RT(S) the CA algorithm gives a better C_{max} than for IT(D)-SP(D)-RT(D). Similarly combination IT(S)-SP(D)-RT(M) gives a better C_{max} than IT(S)-SP(D)-RT(S) and IT(M)-SP(D)-RT(M) gives a better C_{max} than IT(M)-SP(D)-RT(S).
In the group of poorest C_{max} values (with SP(M) value) we can observe that the best C_{max} are achieved while the pair (HR,IH) = (0,1;0,1) - 3 times, while the pair (HR,IH) = (0,5;0,5) - twice and while the pair (HR,IH) = (0,9;0,9) - 4 times; moreover the worst C_{max} is achieved while the pair (HR,IH) = (0,1;0,1) - 3 times, while the pair (HR,IH) = (0,5;0,5) - 4 times and while the pair (HR, IH) = (0,9;0,9) - twice.
In the group of average C_{max} values (with P(S) value) we can obeserve that the best C_{max} is achieved while the pair (HR,IH) = (0,1;0,1) - 3 times, while the pair (HR,IH) = (0,5;0,5) – twice and while (0,9;0,9) – 4 times; moreover the worst C_{max} is achieved while the pair (0,1;0,1) - 4 times, the pair is (0,5;0,5) - 4 times and the pair is (0,9;0,9) - once.
In the group of the best C_{max} values (with SP(D)) we can see that the CA algorithm gives the best C_{max} values while the pair is (0,1;0,1) - twice, (0,5;0,5) - once and at pair (0,9;0,9) – 3 times; moreover the worst C_{max} (with SP(M)) is achieved while the pair is (0,1;0,1) - 4 times, while (0,5; 0,5) - 3 times and while (0,9;0,9) - twice.
Below we present some results achieved when applying higher values of SP, IT and RT than in the main experiment - (ie. SP(V), IT(V), RT(V) values equal 10000).
At SP(V) when the SP increase is from 1000 to 10000 the average value of C_{max} decreases significantly from 69151 to 66060 min. for combination IT(M)-SP(V)-RT(M) compared to IT(M)-SP(D)-RT(M) and from 66859 to 63739 min. for IT(D)-SP(V)-RT(M) compared to IT(D)-SP(D)-RT(M) while the pair (HR,IH) = (0,5;0,5). While the pair (HR,IH) = (0,5;0,5) for combination IT(M)-SP(V)-RT(S) a decrease of C_{max} is achieved from 69836 to 67456 compared to IT(M)-SP(D)-RT(S). For combination IT(S)-SP(V)-RT(S) a decrease of C_{max} is achieved from 67070 to 64626 min. compared to IT(S)-SP(D)-RT(S). Similarly for combination IT(M)-SP(V)-RT(D) a decrease of C_{max} is achieved from 69980 to 67351 compared IT(M)-SP(D)-RT(D), and for combination IT(S)-SP(V)-RT(D) a decrease of C_{max} was achieved from 68057 to 63840 min. compared to IT(S)-SP(D)-RT(S). For combination IT(M)-SP(V)-RT(D) a decrease of C_{max} was achieved from 69980 to 67351 min. compared to IT(M)-SP(D)-RT(D), and for combination IT(S)-SP(V)-RT(D) a decrease of C_{max} - from 68057 to 63840 min. compared to IT(S)-SP(D)-RT(S).
When analizing the influnece of IT(V) on C_{max} we can note that almost in all the analized cases the an icrease from IT(D) to IT(V) gives an insignificant decrease of the C_{max} value eg. for combination IT(V)-SP(D)-RT(M) compared to IT(D)-SP(D)-RT(M), this change is equal to 70800-70407= 393 min.; for combination IT(V)-SP(M)-RT(D) compared to IT(D)-SP(M)-RT(D) the change is equal to 83431-82657 = 774 min.; for combination IT(V)-SP(D)-RT(D) compared to IT(D)-SP(D)-RT(D) the change is equal to 72126-71315 = 811 min.; for combination IT(V)-SP(M)-RT(D) compared to IT(D)-SP(M)-RT(D) the change is equal to 82720-77636= 5084 min.; and for combination IT(V)-SP(S)-RT(D) compared to IT(D)-SP(S)-RT(D) the change is equal to 73647-72 115 = 1432 min.
When analizing the influnece of RT (V) eg. in combinations IT(M)-SP(M)-RT(V) at RT(D) we can observe both increases and decreases of the C_{max} value.
For combinations with the SP(D) value (fig. 5) the CA algorithm has a better C_{max} in all cases as compared to the combinations with the SP(M) value and has a better C_{max} in almost all cases as compared to the combinations with the SP(S) value - as it can be seen in fig. 5.

For combinations with the SP(S) value (fig. 5) the CA algorithm has a better C_{max} in almost all cases as compared to the combinations with the SP(M) value and has a worse C_{max} in all cases as compared to the combinations with SP(D).
For combinations with the SP(M) value (fig. 5) the CA algorithm has a worse C_{max} in almost all cases as compared to the combinations with the SP(S) value and has a worse C_{max} in all cases as compared to the combinations with SP(D).
Overall, the CA algorithm for combinations with the SP(D) value produces solutions of better optimality compared to the CA algorithm for combinations with the SP(S) value and significantly better than with SP(M).
For the problem being solved Gantt charts with the one of best makespan value have been constructed: with machines (Fig. 6), and with parts (Fig.7) while the route is serial.

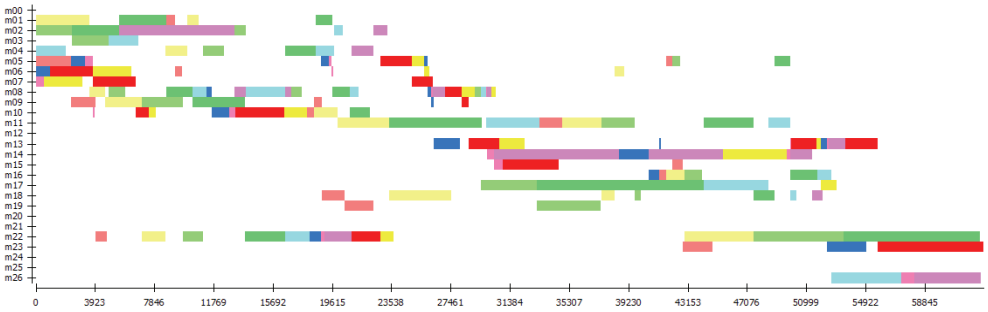


Fig. 6. The Gantt chart for the problem solved for machines (serial route)

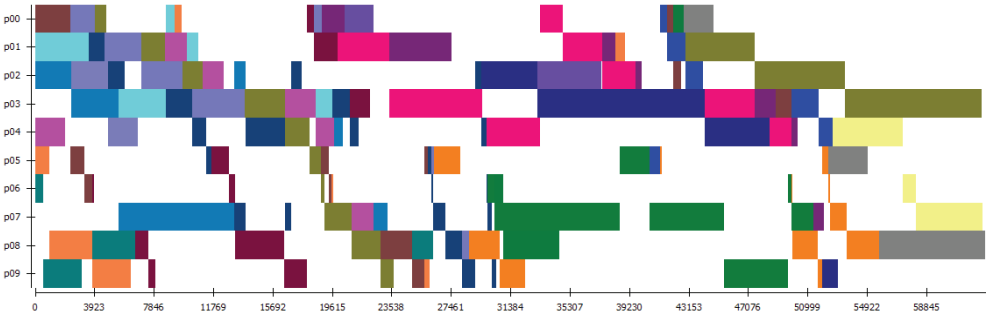


Fig. 7. The Gantt chart for the problem solved for parts (serial routes)

5.2 Comparison of the CA with a genetic algorithm for FJSP

The results obtained with the evolving cellular automata algorithm and genetic algorithm have been compared. A genetic algorithm is characterized by a parallel search of the state space by keeping a set of possible solutions under consideration, called a population. A new generation is obtained from the current population by applying genetic operators such as mutation and crossover to produce new offspring. The application of a GA requires an encoding scheme for a solution, the choice of genetic operators, a selection mechanism and the determination of genetic parameters such as the population size and probabilities of applying the genetic operators.

In our test, we use the genetic algorithm tested in Witkowski et. al (2004, 2007), where there is a more detailed description of the algorithm. Here, we use the recommended parameters, in particular we use a mutation probability of 0.8 and a crossover probability of 0.2.
Figure 8 shows some of the best results for of the CA algorithm, and Table 2 shows some ofthe results for the GA algorithm (serial route).

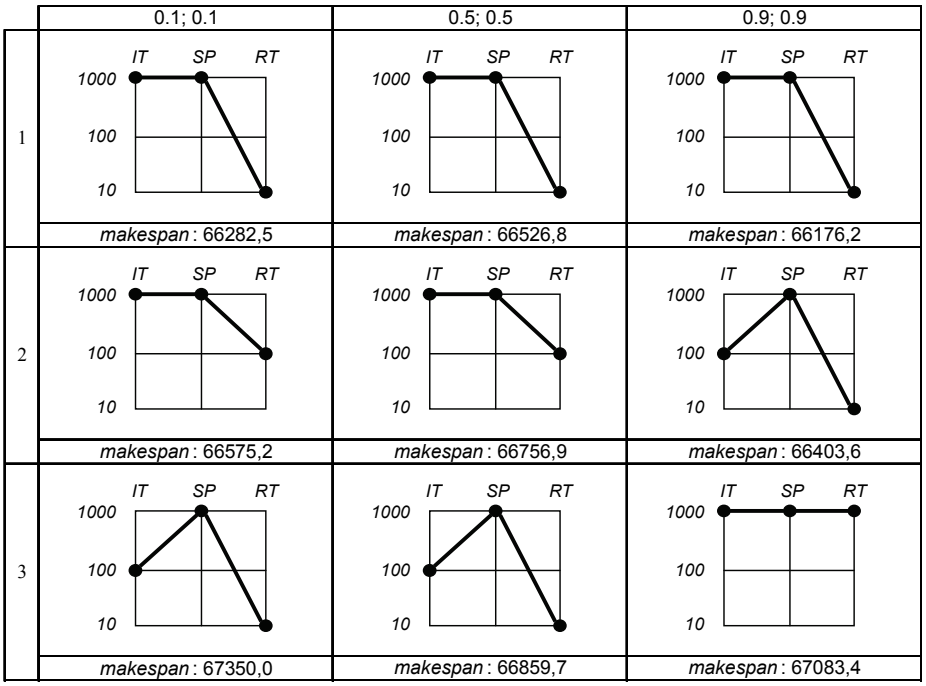


Fig. 8. Some of the best results for the test parameters of CA algorithm (serial route)

	Experiment number	Number of generations	Min imum makespan [min]	Number of schedules	Average makespan [min]
	1	42	59830	595	64380
	2	28	69211	142	74443
	3	44	62664	210	69384
	4	40	67199	120	69230
	5	19	64615	421	69657
	6	6	58734	768	64459
	7	46	63438	330	67457
	8	46	57636	630	61238
	9	33	60236	646	70858
Average		34	62618		67901

Table 2. Some of the results for the GA algorithm (serial route)

In the experiments with the CA algorithm (parallel route) simulations of each test problem were run with the SP population size equal to 10, 100, 1000, the RT transition rate equal to 10, 100, 1000, and the IN iteration number equal to 10, 100, 1000. Each experiment was repeated 10 times.

hybridization ratio = 0.9; intensity of hybridization = 0.9. size of the population = 1000; number of transitions = 10.									
iter.	*Makespan*					*min.*	*average*	*max.*	*avg. time [sec.]*
10	37543.2	38625.2	36427.7	38969.5	37880.1	36427.7	38923.8	41153.0	42
	38976.1	41153.0	39895.2	39895.2	39872.7				
100	37661.2	37281.3	36120.0	37456.5	36597.7	36120.0	37219.7	38143.9	420
	36350.0	37029.1	37433.0	38124.6	38143.9				
1000	36300.6						36330.6		4201

Table 3. Some of the results (with SP (D) value) for the test parameters of the CA algorithm (parallel route)

For the problem being solved Gantt charts with the one of best makespan value have been showed with machines (Fig. 6) while the route is parallel.
In the experiments with the GA algorithm (parallel route) we have used the following: mutation type – single-swap; crossover type - order-based; selection type – roulette. The experiment series was carried out with the following parameters: population size– 1000; generation number - 50. Each experiment was repeated 9 times.
Tabele 4 shows the results for the GA algorithm.

PM 1/	PC 1/	Value of C_{max} for different parameters of the GA algorithm [min]						
1000	1000	**1**	**2**	**3**	**4**	**5**	**6**	**7**
512	128	37545	36724	38155	37585	35637	39430	38822
1000	450	38926	38543	37084	37065	38862	38588	40597
192	256	37368	40725	41188	38902	40709	39457	38531
96	353	37729	38707	40275	37866	39706	39514	39915
256	450	40018	35124	39795	38777	37631	38515	38777
64	256	40359	38339	36397	37939	38109	38610	39853
128	64	39775	38181	40018	37210	40112	39615	38968
16	256	41088	38055	40519	40566	39857	37301	39629
8	128	40200	41239	39281	40422	39025	40047	41556
8	256	40942	38210	38390	40132	39879	39890	35470
8	450	36868	39628	39560	39932	39959	38842	40078
32	450	37158	40589	40653	37440	37855	38031	39183
32	256	39961	38647	41262	40165	39269	35166	39483
32	128	40567	40193	39226	39579	40423	38843	39825
64	128	42941	38274	39981	39491	40111	39908	37428

64	256	38145	38874	36901	39209	38915	40416	39649
64	450	39013	39672	39344	40236	39826	39775	41053
128	353	38671	38903	39186	39422	37540	36506	38223
512	353	40968	39619	40180	39583	38870	40559	36429
192	256	37466	39257	38295	38501	38132	40304	39477
96	256	39444	38595	38905	38402	38774	38134	39817
64	353	37770	38677	39667	40263	39317	37676	39445
16	353	39930	39980	41221	38075	38681	39236	40329
280	130	40586	39648	36805	38176	38239	36077	39873
280	370	39253	38892	38222	38726	38130	40813	39168
840	130	41824	37050	33949	37923	37435	38013	38712
910	370	37426	37692	38223	38467	37562	40862	39039
8	64	38436	41069	37987	39287	38690	39993	39042
16	64	37505	38351	36647	39846	34738	40428	39065
32	64	38554	39284	38144	38171	41728	39474	38144
64	64	40415	39862	36583	38454	36264	39039	40426
96	64	40342	39791	39055	39394	39872	38728	37385
192	64	40652	36435	38656	40164	38549	37849	37345
256	64	40814	38851	40186	40448	39435	41031	37419
280	64	38278	38972	38653	38102	37330	38661	39112
512	64	38788	37774	39794	38820	40603	39224	39700
840	64	40521	38146	37848	38323	38501	38126	37391
910	64	39983	39467	38609	35735	38880	37447	41173
1000	64	38465	40961	36692	38726	39597	38493	37194
16	128	38645	38577	39785	39314	39181	39658	37034
96	128	40260	39932	39688	39593	39786	38340	38967
128	128	39873	39150	37624	39528	40273	38549	40675
192	128	38559	39003	36695	40691	38635	39235	39041
840	128	38322	39987	37723	40669	39489	38108	39049
910	128	37698	39438	37920	35857	40082	39579	34667
1000	128	38599	39670	39822	37052	38917	38312	40619
8	130	38254	39256	37131	37520	40000	39848	39644
16	130	41453	38340	37480	40308	37198	40346	39298
32	130	38986	39323	39196	39646	40179	38961	38651
64	130	38597	38765	39151	40179	39599	38224	40713
96	130	41377	38974	40071	39615	38247	40970	39434
128	130	40669	40836	39114	40808	39476	37848	38918
192	130	38561	39158	39877	39820	39020	38387	40398
256	130	36108	37828	38378	40828	38998	40361	38449
512	130	38935	39359	37933	33549	37184	40117	39846

910	130	38595	40498	40037	39457	38055	37419	39086
1000	130	38604	37872	39596	39220	39818	38110	39327
128	256	35614	38756	37070	40614	40770	38541	39979
256	256	39891	39413	35647	38149	38485	41064	40019
280	256	38330	36906	37492	39298	39913	39255	39768
512	256	39951	39321	39691	37334	36105	36620	40355
840	256	38195	39439	39945	40077	40545	38156	36714
910	256	39265	40465	37356	40592	39022	31701	38760
1000	256	41797	40467	40199	37534	37890	39089	38813
8	353	38198	38215	39587	40206	37049	37457	37841
32	353	38420	38526	38422	36501	39419	37903	39557
192	353	39418	39418	35429	39043	39223	40586	39113
256	353	39693	38591	39548	37636	39295	39853	38624
280	353	37849	35995	39018	38860	37544	38312	37608
840	353	37159	37159	39290	39942	38015	37159	38966
910	353	41668	34752	39535	37406	39493	37733	38185
1000	353	39868	40506	38434	39869	38237	39632	38582
8	370	40255	38051	39140	38494	40274	38577	40698
16	370	39564	40285	39366	39737	39077	36294	40451
32	370	39253	40837	39576	37373	41460	38928	36018
64	370	38105	38778	39236	39644	36098	38552	40660
96	370	36646	39237	38242	36963	40197	38688	39378
128	370	39074	39184	38297	39650	40936	37823	38687
192	370	39305	36835	40083	41108	39513	38629	38312
256	370	38860	41255	42417	39995	34023	40120	37957
512	370	39695	40118	38358	39824	40379	40349	39603
840	370	40084	38184	37808	38206	37710	40083	38468
1000	370	38875	38661	40703	36498	38493	38535	37560
16	450	38756	39286	38315	39106	38933	40246	35516
96	450	35559	38088	39556	38268	40046	37104	40137
128	450	38532	40352	38448	42062	39222	38763	39441
192	450	39985	41215	40856	36504	40110	35746	39633
280	450	37565	38601	39134	39571	41389	38153	37565
512	450	38871	38547	38282	39381	39058	37768	37832
840	450	39854	38626	39132	33809	38059	40107	32947
910	450	36160	39825	38749	38673	39747	37460	39459

Tabele 4. Results (value C_{max}) for different parameters of the GA algorithm

The experiments showed that for CA algorithm we can achieve results similar to the GA algorithm – both for the serial route and parallel routes.

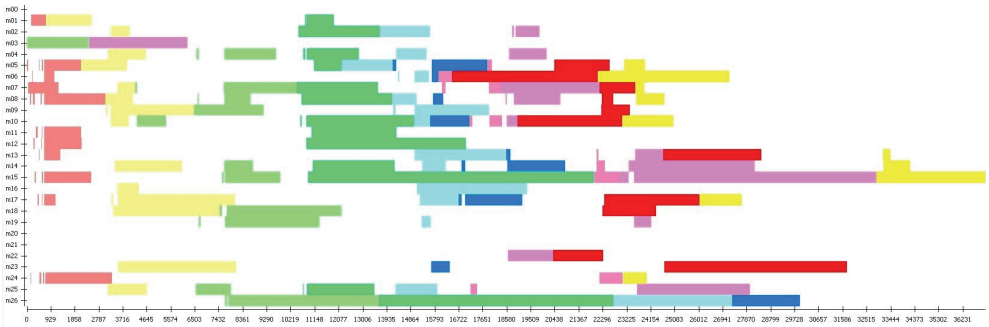


Fig. 9. Gantt chart for the solving problem with parallel route (for machines)

6. Conclusion

The paper presents an algorithm based on evolving cellular automata for solving flexible job shop scheduling problem. The presentation of the algorithm CA and its comparison with the GA algorithm shows positive results. The software of this algorithm allows for analysis of the schedule construction process for many variants reflecting a variety of combinations of other factors. We can generally see that depending on the PS population size we can single out 3 classes of quality results with regard to the C_{max} criterion - very good (large population size), average (medium population size) and poor (small population size). Moreover an increase of the IT value influences the C_{max} more than the RT value, although there are a number of exceptions..

In addition, we observed that for our specialized FJSP problem the trajectory methods (e.g. tabu search, simulated annealing, GRASP) have better efficiency than the CA algorithm, particularly when those algorithms are used in hybrid approaches [Witkowski et al., 2005a, 2005b, 2006]. Experiments for the analyzed FJSP problem indicate that the evolving cellular automata algorithm is comparable with such population-based methods as the genetic algorithm. Morover, the successful use of this approach will also depend on the amount of calculation that can be done and on further improvement of this algorithm for our problem.

7. References

Fattahi,P., Mehrabad, M.S. & Jolai, F. (2007). Matemathical modeling and heuristic approaches to flexible job shop scheduling problems. *Journal Intel. Manufacturing*, Vol. 18, pp. 331-342

Kacem, I., Hammandi, S. & Borne, P. (2002). Approach by localization and multiobjective evolutionary optimization for flexible job shop scheduling problems, *IEEE T. System Man Cybernetics C.*, Vol. 32, pp.1-13

Liu, H., Abraham, A., Choi, O. & Moon, S.H. (2006). Variable Neighborhood Particle Swarm Optimization for Multi-objective Flexible Job-Shop Scheduling Problems, In T.-D. Wang et al. (eds), SEAL 2006, *LNCS* 4247, pp. 1997-2004.

Ong, Z.X., Tay, J.C. & Kwoh, C.K. (2005). Applying the Clonal Selection Principle to Find Flexible Job Shop Schedules ", ICARIS 2005, LNCS 3627, pp. 442-455

Preissl, R. (2006). A Parallel Approach For Solving The Flexible Job Shop Problem With Priority Rules Developed By Genetic Programming, Master's thesis, J. Kepler University, Linz

Shi-Jin, W., Bing-Hai, Z. & Li-Feng, X. (2008). A filtered-beam-search-based heuristic algorithm for flexible job shop scheduling problem. *International Journal Production Research*, Vol. 46, pp. 3027-3058

Tay, J.C.& Ho, N.B. (2008). Evolving dispatching rules using genetic programming for solving multi-objective flexible job shop problems", *Comput. Ind. Eng*., Vol. 54, pp. 453-473

Yazdani, M., Gholami, M., Zandieh, M. & Mousakhani, M. (2009). A simulated Annealing Algorithm for Flexible Job Shop Scheduling Problem, *Journal of Applied Sciences*, pp. 1-9

Affenzeller, M. & Wagner, S. (2004). SASEGASA: A New Genetic Parallel Evolutionary Algorithm for Achieving Highest Quality Results. *Journal of Heuristics-Special Issue on New Advances on Parallel MetaHeuristics for Complex Problem,* Vol. 10, pp. 239-2630

Back, D., Fogel, B. & Michalewicz Z. (eds.) (1997). *Handbook of Evolutionary Computation,* Oxford University Press and Institute of Physics Publishing, Bristol-NY

Beham, A., Winkler, S., Wagner, S. & M. Affenzeller, M. (2008). A Genetic Programming Approach to Solve Scheduling Problems with Parallel Simulation. *Parallel and Distributed Processing*, pp. 1-5.

Koza, J.R. (1992). *Genetic Programming: On the Programming of Computers by Natural Selection*, MIT Press, Cambridge

Mitchell, M., & Forrest, S. (2005). *Genetic Algorithms and Artificial Life, Artificial Life. An Overview.* , G. Langton (Ed.), MIT Press, 1995.

Mitchell, M., Crutchfield, J. & R. Das, R. (1996). Evolving Cellular Automata with Genetic Algorithms: A Review of Recent Work. *Proceedings of the First International Conference on Evolutionary Computation and Its Applications* , 1996.

Zomaya, A. Y., Ward, C. & Macey, B. (1999). Genetic Scheduling for Parallel Processor Systems: Comperative Studies and Performance Issues, *IEEE Trans. on Parallel and Distributed Systems,* Vol. 10, N 8, pp. 795-812

Stocher, W., Kabelka, B., & Preissl, R. (2007). Automatically Generating Priority Rules for the Flexible Job Shop Problem with Genetic Programming", *Proc. of Computer Aided Systems Theory*, Euro CAST 2007

Winkler, S., Affenzeller, M. & Wagner, S. (2007). Advanced Genetic Programming Based Machine Learning. *Journal of Mathematical Modelling and Algorithms*, Vol. 6, N 3, pp. 455-480

De Castro, L. N. (2006). *Fundamentals of Natural Computing: Basic Concepts, Algorithms, and Applications*, Chapman&Hall/CRC, NY.

Tomassini, M., Sipper, M. & Perrenoud, M. (2000). On the Generation of High-Quality Random Numbers by Two-Dimensional Cellular Automata, *IEEE Trans. Computers*, Vol. 49, N 10, pp. 1140-1151

Seredyński, F. & Święcicka, A. (2002). Immune - like System Approach to Cellular Automata – based Scheduling, *Parallel Processing and Applied Mathematics*, R. Wyrzykowski et al. (Ed.), LNCS 2328, pp. 626-633, Springer Berlin/ Heidelberg

Seredyński, F. & Zomaya A. Y. (2002). Sequential and Parallel Cellular Automata-Based Scheduling Algorithms. IEEE Trans. Parallel Distributed Systems 13(10), pp. 1009-1023

Back, T., & Breukelaar, R. (2005). Using Genetic Algorithms to Evolve Behavior in Cellular Automata, In: *Lecture Notes in Computer Science*, Springer Berlin/ Heidelberg, vol. 3699, pp. 1-10

Kanoh, H., Wu, Y. (2003). Evolutionary Design of Rule Changing Cellular Automata, In: Palade, V., Howlett, R.J., Jain, L.C (Eds.). Knowlegde-Based Intelligent Information and Engineering Systems, 7th International Conference KES 2003, Oxford, UK, September 3-5, 2003, Lecture Notes in Computer Science, Springer Berlin/Haidelberg, Vol. 2773, pp. 258-264

Martins, C. L. M. & de Oliveira, P.P.B. (2005). Evolving Sequential Combinations of Elementary Cellular Automata Rules, In: Capcarrere, M. et al. (Ed.), LNAI 3630, pp. 461-470

Das, R., Mitchell, M &J. P. Crutchfield, J. P. (1994). A Genetic Algorithm Discovers Particle-Based Computation in Cellular Automata, In: Parallel Problem Solving from Nature – PPSN III, Davidor, Y. et al. (Ed.), LNCS 866, Springer, pp. 344-355

Sipper, M. (1997). *Evolving of Parallel Cellular Machines: The Cellular Progrmming Approach*, Springer-Verlag, Heidelberg

Sipper, M. (1999). The Emergence of Cellular Computing, IEEE Computer, Vol. 32, N 7 , pp. 18-26, July 1999

Subrata, R., & Zomaya, A. Y. (2003). Evolving Cellular Automata for Location Management In Mobile Computing, *IEEE Trans. on Parallel and Distributed Systems*, Vol. 14, N 1, pp. 13-26

Sahoo, G. & Kumar, T. (2007). A Genetically based Evolutionary Computing Technique based on Cellular Automata, *International Journal of Computer Science and Network Security*, Vol. 7, N 11, pp. 26-31

Witkowski, T., Antczak, A. & Antczak, P. (2004). Random and Evolution Algorithms of Tasks Scheduling and the Production Scheduling, *Proceedings of International Join Conference on Fuzzy Systems – IEEE 2004*, Budapest, July 2004 vol. 2, pp. 727-732

Witkowski, T., Antczak, P. & Antczak, A. (2005 a). Tabu Search and GRASP used in hybrid procedure for optimize the flexible job shop problem, In: Fuzzy Logic, Soft Computing and Computational Intelligence - *11th* IFSA World Congress, Liu, Y. et al. (Ed.), Tsinghua University Press and Springer, pp. 1620-1625

Witkowski, T., Antczak P.& Antczak A. (2005 b). Application of GRASP procedure for production scheduling and its comparision with other methods. *Journal of Automation and Information Sciences*, Beggel House Inc., New York, Vol. 37, 6(40), pp. 35-40

Witkowski T., Antczak, P. & Antczak, A. (2006). The application of simulated annealing procedure for the flexible job shop scheduling problem, *Proceedings of 11th International Conference.: Information Processing and Management of Uncertainty in Knowledge-Based Systems (Industrial Track),* IPMU 2006, Paris, 2006, pp. 21-26

Witkowski, T., Elzway, S., Antczak, A. & Antczak, P. (2007). Representation of Solutions and Genetic Operators for Flexible Job Shop Problem, In (eds. D.-S. Huang, L. Heutte, and M. Loog): ICIC 2007, CCIS N2: Advanced Intelligent Computing Theories and Applications. Springer-Verlag, Berlin Heidelberg 2007, pp. 256- 265

Witkowski, T., Antczak, A., Elzway, S., Antczak, P. (2009). Evolving Cellular Automata - based Flexible Job Shop Scheduling, 5th International Conference on Natural Computation ICNC'09, Vol. 2 (eds. H. Wang, K. S. Low, K. Wei, & J. Sun); Tianjian, China, 14-16 August 2009, CPS, Los Alamitos, California, Washington, Tokio, pp. 8-13

Part 4

Statistical Physics and Complexity

19

Nonequilibrium Phase Transition of Elementary Cellular Automata with a Single Conserved Quantity

Shinji Takesue
Kyoto University
Japan

1. Introduction

Conservation laws are one of the most important concepts in physics. In a Hamiltonian dynamical system, the state of which is represented by a point in the phase space, energy is conserved and the motion of the system is restricted on a surface where energy is a constant or, if more conserved quantities exist, a submanifold of lower dimensions. If the system has as many conserved quantities as the degrees of freedom, it is integrable and the motion of the system is completely understood. In many cases, the existence of a conserved quantity is connected with some symmetry by Noether's theorem. There are many cases where finding a hidden symmetry and the corresponding conservation law deepen the understanding of a physical system. Contrastingly, nonexistence of redundant conserved quantities is important in statistical mechanics. It is because statistical mechanics is constructed on the three hypotheses; preservation of phase space volume (Liouville's theorem), energy conservation, and ergodic property of the motion, and the last one implies that the system does not have a conserved quantity except energy and mass. In dissipative systems, energy is no more conserved but the mass or the number of particles is still conserved in many interesting systems including granular matter.

Considering the wide applications of cellular automata to physics, it is natural to study conserved quantities in cellular automata (CA). Hattori and the author gave a necessary and sufficient condition for a CA to have an additive conserved quantity which is a sum of local quantities and whose value does not change under the time evolution of CA(Hattori and Takesue 1991). Exact definition of the additive conserved quantity will be given in Section 2. In that paper, the condition was applied to Wolfram's elementary CA (ECA) and their reversible variants, elementary reversible CA (ERCA) and the table of additive conserved quantities for every rule was obtained. Generalization to staggered quantities was done in (Takesue 1995). These works clarified that the number of conserved quantities depends on rules. Moreover, it is often the case that not only the sum but also each summand itself is conserved(Takesue 1989). It corresponds to class 2 in Wolfram's classification, which is characterized by separated simple stable or periodic structures. In the light of physics, we are more interested in such a CA that has a small number of additive conserved quantities whose summand is not invariant.

Dynamical behavior of the conserved quantities is very different between reversible and irreversible CA. Reversible CA satisfy the preservation of the phase space volume owing to the discreteness of states. Thus, by regarding an additive conserved quantity as a Hamiltonian, we can construct Gibbs statistical mechanics on the CA. Then its connection with dynamics is an interesting subject. The formation of the canonical distribution in a subsystem (Takesue 1987), Fourier's law and Kubo formula for the thermal conduction (Takesue 1990a), relaxation to equilibrium (Takesue 1990b), and Boltzmann-type equations (Takesue 1997) were discussed for ERCA. In particular, two rules (rules 26R and 94R) were found to show diffusive behavior in the macroscopic scale. Furthermore, we can devise two-dimensional CA which conserve the Ising Hamiltonian with or without other degrees of freedom(Creutz 1985; Vichniac 1984). Using those CA, dynamics at the critical point was studied (Saito et al. 1999).
Concerning irreversible CA, additive conserved quantities should be considered as the particle number rather than energy. In this context, number-conserving CA, where the number of 1s is conserved, are well studied. One of the interesting behavior is density classification. Originally, density classification problem meant searching a CA that has the following property: If an initial density of 1s exceeds a given threshold, the CA evolves into the state of all 1s, and if the density is below the threshold, the state of all 0s is reached. A number of rules were proposed, but finally it was proved that perfect classification is impossible for one-dimensional two-state CA(Land and Belew 1995). However, M. S. Capcarrere et al found that ECA rule 184 in Wolfram's notation performs the classification if the output condition is loosened(Capcarrere 2001). Rule 184 is a number-conserving CA. Thus, there must be a block 11 if the density of 1s is above $1/2$ and a block 00 if the density is below $1/2$. Because rule 184 tends to place 0 and 1 as alternatingly as possible, blocks 11 disappear and one or more block 00 remain below $1/2$ by the time $N/2$, where N is the system size. Therefore the presence of a block 00 after $N/2$ indicates that the density is below $1/2$. Similarly, blocks 11 indicates the density is above $1/2$. Note that the behavior is observed under the cyclic boundary condition. In a recent paper(Takesue 2008), the author showed that ECAs with a single additive conserved quantity classify the density of the conserved quantity. Moreover, the paper showed preliminary results that the same rules can show a kind of nonequilibrium phase transition when some stochastic boundary conditions are employed. In this chapter, we will focus on the latter phenomenon in the ECAs.
In the next section, the additive conserved quantities are defined and the conservation condition is derived. The condition is applied to the ECAs to find the rules with a single additive conserved quantity. The nonequilibrium phase transition discussed here is originally found in a continuous Markov chain called the asymmetric simple exclusion process (ASEP). In Section 3, the ASEP and its phase transition are introduced and the mechanism of the phase transition is clarified by the so-called domain wall theory. We discuss what stochastic boundary condition is suited for the ECAs in Section 4. In Section 5, the probability distribution of patterns are calculated and the domain wall theory is applied to the ECAs. Section 6 describes diffusive behavior of the domain wall observed just on the phase transition line. The last section is devoted to discussion and conclusion.

2. Elementary CA and conserved quantities

Wolfram' s elementary cellular automata (ECA)(Wolfram 1983) are a class of one-dimensional cellular automata with two possible states for each cell and local update rules which depend only on three neighbor cells. That is, if $x_i^t \in \{0,1\}$ denotes the value of cell i at time t, the

evolution of an elementary CA is written as

$$x_i^{t+1} = f(x_{i-1}^t x_i^t x_{i+1}^t) \tag{1}$$

with some fixed function $f : \{0,1\}^3 \to \{0,1\}$. Thus there are $2^{2^3} = 256$ different rules in the ECAs, but the rules that are transformed into each other by the left-right inversion or the exchange of 0 and 1 or their composite are isomorphic and accordingly the ECA are classified into 88 equivalence classes.

Let us impose the cyclic boundary condition of period N on the ECA and denote the configuration at time t by $x^t = (x_0^t x_1^t \dots x_{N-1}^t)$. Now we consider a function of x^t of the form

$$\Phi(x^t) = \sum_{i=0}^{N-1} E(x_i^t x_{i+1}^t \dots x_{i+k}^t) \tag{2}$$

where $E(x_0 x_1 \dots x_k)$ is some function of $k+1$ variables. If $\Phi(x^1) = \Phi(x^0)$ holds for any $x^0 \in \{0,1\}^N$, this quantity Φ is called an additive conserved quantity of range k and the corresponding E the conserved density.

In (Hattori and Takesue 1991) a necessary and sufficient condition for E to be a conserved density was derived as follows. First, assume that Φ is an additive conserved quantity of range k. Then, for any $x = (x_i)$, the following equality must hold:

$$\sum_{i=0}^{N-1} [G(x_i x_{i+1} \dots x_{i+k+2}) - E(x_i \dots x_{i+k})] = 0, \tag{3}$$

where G is the function of $k+3$ variables defined as

$$G(x_0 x_1 \dots x_{k+2}) = E(f(x_0 x_1 x_2) f(x_1 x_2 x_3) \dots f(x_k x_{k+1} x_{k+2})) \tag{4}$$

and indices are understood mod N. Equality (3) holds if we assume $x_0 = 0$.

$$\sum_{i=0}^{N-1} [G(x_i x_{i+1} \dots x_{i+k+2}) - E(x_i \dots x_{i+k})] \Big|_{x_0=0} = 0. \tag{5}$$

Subtraction of Eq. (5) from Eq. (3) leaves only an N-independent number of terms.

$$\sum_{i=0}^{k+2} [G(x_{i-k-2} \dots x_i) - G(x_{i-k-2} \dots x_{-1} 0 x_1 \dots x_i)] + \sum_{i=0}^{k} [E(x_{i-k} \dots x_i) - E(x_{i-k} \dots x_{-1} 0 x_1 \dots x_i)] = 0 \tag{6}$$

We can further put $x_{-k} = x_{-k+1} = \dots = x_{-1} = 0$ and utilize $G(00\dots0) = E(0\dots0)$ to obtain the following equality,

$$\begin{aligned} G(x_0 x_1 \dots x_{k+2}) - E(x_0 x_1 \dots x_k) = & \sum_{i=0}^{k+1} [-G(0\dots0 x_0 x_1 \dots x_i) + G(0\dots0 x_1 x_2 \dots x_{i+1})] \\ & + \sum_{i=1}^{k} [E(0\dots0 x_0 x_1 \dots x_{i-1}) - E(0\dots0 x_1 x_2 \dots x_i)]. \end{aligned} \tag{7}$$

Clearly, this is a necessary condition for E to be a conserved density. In fact, it is also a sufficient condition, because Eq. (7) can be rewritten in the form of equation of continuity

$$E(x_i^{t+1} x_{i+1}^{t+1} \dots x_{i+k}^{t+1}) - E(x_i^t x_{i+1}^t \dots x_{i+k}^t) = J(x_{i-1}^t x_{i+}^t \dots x_{i+k}^t) - J(x_i^t x_{i+1}^t \dots x_{i+k+1}^t), \tag{8}$$

where current function J is defined by

$$J(x_0x_1\dots x_{k+1}) = \sum_{i=0}^{k+1} [E(0\dots 0x_0x_1\dots x_{i-1}) - G(0\dots 0x_0x_1\dots x_i)]\,. \tag{9}$$

Therefore, Eq. (7) is a necessary and sufficient condition, which is called the Hattori-Takesue condition.
It is evident that $E(x_0x_1\dots x_k) = S(x_0\dots x_{k-1}) - S(x_1\dots x_k)$ leads to $\Phi(x) = 0$ for any function $S(x_0\dots x_{k-1})$. To remove such trivial solutions from Eq.(7), we can assume that $E(0x_1\dots x_k) = 0$ for any $(x_1\dots x_k)$, and then the conservation condition is simplified as

$$G(x_0\dots x_{k+2}) - E(x_0\dots x_k) = \sum_{i=0}^{k+1} [G(0\dots 0x_1x_2\dots x_{k+2-i}) - G(0\dots 0x_0x_1\dots x_{k+1-i})]\,. \tag{10}$$

In some cases, however, E with the condition $E(0x_1\dots p_k) = 0$ is not convenient for use and adding some surface term $S(x_0\dots x_{k-1}) - S(x_1\dots x_k)$ to it is preferable. Solutions of the equations (10) forms a vector space in the function space and we refer to its dimension as the number of additive conserved quantities of range k. This number increases with k, because the additive conserved quantities of range k include those of smaller ranges.
The conservation condition was generalized to staggered invariants, where factor $(-1)^i$ and/or $(-1)^t$ is introduced in the rhs of Eq.(2) (Takesue 1995). The numbers of additive and staggered conserved quantities of range $k = 6$ were listed for all 88 equivalence classes in that paper and those of $k = 9$ in (Takesue 2008). The numbers of conserved quantities depend on the rules. Some rules do not have a conserved quantity, some others have more than one, and the rules 11, 14, 35, 43, 56, 142, and 184 (and the rules isomorphic to them) have only one conserved quantity for each. Namely, each of them has a single additive conserved quantity, whose density E is not conserved, and no staggered invariants. Their respective conserved densities are listed in Table 1.

Rules	Conserved density
184	$E(x) = x$
14, 35, 43, 142	$E(xy) = (x-y)^2$
56	$E(xyz) = x+y+z-3xyz$
11	$E(xyzw) = x(1-y)[1-z(1-w)]$

Table 1. ECA rules with a single conserved quantity and their conserved densities

3. Phase transition in ASEP

The nonequilibrium phase transition which we will discuss was originally found in stochastic particle systems on a lattice. The famous example is the asymmetric simple exclusion process (ASEP) with open boundaries(Derrida et al. 1993; Sasamoto 1999**?**). It resembles ECA in one dimension and two possible states of a cell but time is continuous and the dynamics is stochastic. Consider a one-dimensional lattice composed of N cells. Each cell i is occupied by a particle ($\tau_i = 1$) or empty ($\tau_i = 0$). During an infinitesimal time interval dt each particle can hop to the right neighbor with probability dt and to the left neighbor with probability qdt, where q is a nonnegative number less than 1, provided that the destination is empty. Moreover, a particle is added to cell 1 with probability αdt if the cell is empty. Similarly, a particle is removed from cell N with probability βdt if the cell is occupied. From any initial

condition, the system goes to a steady state after some relaxation time. Since the steady state has a nonzero rightgoing current, it represents a nonequilibrium steady state with particle flow. The probability distribution of the system in the steady state was exactly obtained for the case $q = 0$, which is called the totally asymmetric exclusion process (TASEP), by Derrida et al(Derrida et al. 1993) using the method of matrix products. The result was extended to $q \neq 0$ by Sasamoto(Sasamoto 1999) and to the case where a particle can enter or exit from both ends by Uchiyama et al(Uchiyama 2004). It is remarkable that the ASEP shows phase transitions depending on the parameters for the boundary condition. Figure 1 shows the phase diagram of the TASEP. Region A ($\alpha < 1/2$ and $\alpha < \beta$) represents the low-density phase where the particle density is $\langle \tau_i \rangle \simeq \alpha$ and the current is $\alpha(1-\alpha)$. Region B ($\beta < 1/2$, $\alpha > \beta$) is the high-density phase where the density is $\langle \tau_i \rangle \simeq 1-\beta$ and the current is $\beta(1-\beta)$. Region C ($\alpha > 1/2$, $\beta > 1/2$) is the maximal current phase where the density is $1/2$ and the current is $1/4$. The density changes discontinuously across the line $0 < \alpha = \beta < 1/2$. Thus, it is called the line of the first-order phase transition. The transitions between regions A and C and between B and C are the second-order, because the change of density is continuous. Similar phase diagrams are obtained for the general ASEP.

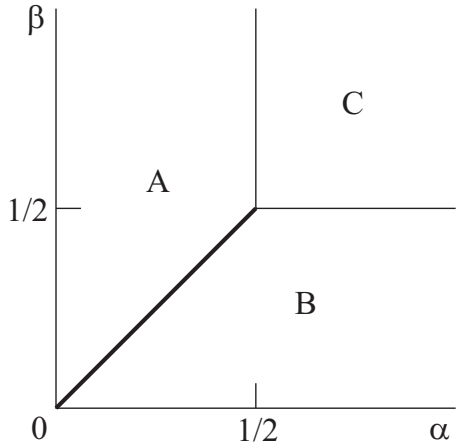


Fig. 1. The phase diagram of the TASEP. There are three phases: low-density phase (A), high-density phase (B) and the maximal current phase (C). The phase transition between the low and high density phases is the first-order and the particle density changes discontinuously. The phase transition between A and C and that between B and C are the second-order and the density changes continuously.

The mechanism of the phase transition is understood in terms of the behavior of a phase boundary. This is called the domain wall theory(Kolomeisky et al. 1998). Now we do not see the 0-1 sequence or discrete-time dynamics but consider coarse-grained density variation of the ASEP. That is, dynamical behavior of a cell is replaced by continuous change of locally averaged density profile. After some transient relaxation process, a part of the system can be regarded as belonging to one of the three phases. Let us assume that the system is composed of two different phases and that the left phase has density ρ_L and current J_L and the right phase has density ρ_R and current J_R. Then the velocity of the phase boundary (domain wall), V, is obtained in the same manner as the theory of shock wave in fluid mechanics and the

result is

$$V = \frac{J_R - J_L}{\rho_R - \rho_L}. \tag{11}$$

This is an outcome of the conservation law. If V is positive, the domain wall goes to the right end and the left phase prevails in the system. Conversely, if V is negative, the right phase dominates the system. Accordingly, the phase transition occurs when $V = 0$. This argument applies to the TASEP. Assume that the left phase is the low-density phase and the right one is the high-density phase. Then, $V = \beta - \alpha$. This means that the transition occurs at $\alpha = \beta$. The transition lines between the maximal current phase and the other phases are also successfully explained in the similar manner.

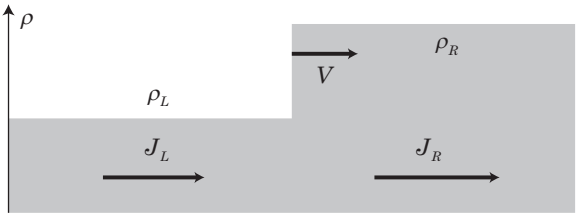


Fig. 2. Velocity of the domain wall is determined by the densities and currents in the two phase.

4. Stochastic boundary condition for the ECA

We will see that the ECAs with a single additive conserved quantity exhibit nonequilibrium phase transition of the same type as in the ASEP if an appropriate open boundary condition is employed. The first example is rule 184. This rule conserves the number of 1s as the ASEP does. Thus the following stochastic evolution is naturally devised. Consider the system of $N + 2$ cells, which are numbered from 0 through $N + 1$. In the evolution from time t to $t + 1$, the states of the cells 1 through N are updated according to the rule (1). For cells 0 and $N + 1$, the states are chosen with probability as

$$x_0^{t+1} = \begin{cases} 1 \text{ with probability } \alpha \\ 0 \text{ with probability } 1-\alpha \end{cases}, \quad x_{N+1}^{t+1} = \begin{cases} 1 \text{ with probability } 1-\beta \\ 0 \text{ with probability } \beta. \end{cases} \tag{12}$$

This is equivalent to the evolution of probability distribution as

$$p^{t+1}(x) = p_L(x_0)p_R(x_{N+1}) \sum_{x'_0,\ldots,x'_{N+1}} \prod_{i=1}^{N} \delta(x_i, f(x'_{i-1}x'_ix'_{i+1}))p^t(x') \tag{13}$$

where $p^t(x)$ denotes the probability that $x^t = x = (x_i)_{0\le i\le N+1}$, $\delta(x, y)$ is Kronecker's delta $\delta(x, y) = 1$ if $x = y$ and $\delta(x, y) = 0$ if $x \neq y$, $p_L(u) = \alpha u + (1-\alpha)(1-u)$ and $p_R(v) = (1-\beta)v + \beta(1-v)$. For various α and β, we numerically computed time averages of the density of 1s, $\rho = N^{-1}\sum_{i=1}^{N}\langle x_i\rangle$, and current $J = (N-1)^{-1}\sum_{i=1}^{N-1}\langle x_i(1-x_{i+1})\rangle$ in the steady states, where $\langle\ \rangle$ represents the time average. The result is shown in Fig. 3. Just as in the ASEP, region $\alpha < \beta$ is the low-density phase, where $\rho = \alpha$, $J = \alpha(1-\alpha)$ and region $\alpha > \beta$ is the high-density phase, where $\rho = 1-\beta$, $J = \beta(1-\beta)$. The first order phase transition occurs at line $\alpha = \beta$. Note that the maximal current phase does not appear in the ECA.

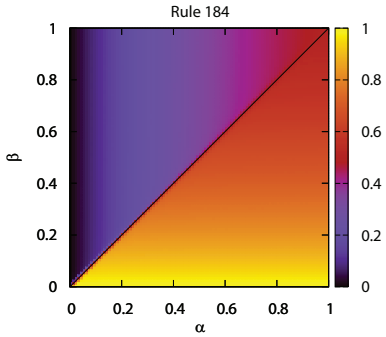


Fig. 3. Phase diagrams of rule 184 with the stochastic boundary condition (13). The average density of 1s is illustrated. The phase transition line $\alpha = \beta$ is also depicted.

We can adapt the boundary condition to the ECA with conserved density $E(xy) = (x-y)^2$ as

$$x_0^{t+1} = \begin{cases} 1 - x_1^{t+1} \text{ with probability } \alpha \\ x_1^{t+1} \text{ with probability } 1-\alpha \end{cases}, \quad x_{N+1}^{t+1} = \begin{cases} 1 - x_N^{t+1} \text{ with probability } 1-\beta \\ x_N^{t+1} \text{ with probability } \beta \end{cases} \tag{14}$$

This is equivalent to the following evolution of probability distributions.

$$p^{t+1}(x) = p_L(x_0|x_1)p_R(x_{N+1}|x_N) \sum_{x'_0,\ldots,x'_{N+1}} \prod_{i=1}^{N} \delta(x_i, f(x'_{i-1}x'_ix'_{i+1}))p^t(x'), \tag{15}$$

where the conditional probabilities ρ_L and ρ_R are defined as

$$p_L(u|v) = \alpha\delta(E(uv),1) + (1-\alpha)\delta(E(uv),0), \quad p_R(u|v) = (1-\beta)\delta(E(vu),1) + \beta\delta(E(vu),0). \tag{16}$$

As shown in Table 1, the rules with the conserved density are rules 14, 35, 43 and 142. However, rule 43 is equivalent to rule 184 by block transformation 00, $11 \to 0$ and 01, $10 \to 1$. Namely, if we denote the transformation by b, the following equality holds:

$$b\left(f_{43}(x_0x_1x_2)f_{43}(x_1x_2x_3)\right) = f_{184}\left(b(x_0x_1)b(x_1x_2)b(x_2x_3)\right), \tag{17}$$

where f_{43} and f_{184} are respective rule functions. Thus, rule 43 of size $N+2$ with the stochastic boundary condition (15) is transformed into rule 184 of size $N+1$ with the stochastic boundary condition (13) and accordingly exhibits the same nonequilibrium phase transition. Similarly, rule 142 is transformed into rule 184 by another block transformation 01, $10 \to 0$ and 00, $11 \to 1$. In this case, the transformation is accompanied with change of parameter values $\alpha \to 1-\alpha$, $\beta \to 1-\beta$. The phase transition of the same type occurs in this rule also. Rules 14 and 35 are not equivalent to rule 184 and are considered in the next section. We only mention here that the current function for rule 14 is $J(xyz) = -xy - (1-x)(1-y)z$, and that for rule 35 is $J(xyz) = x(1-y)(1-z)$.

For rules 56, we must have further consideration. The additive conserved quantity for this rule has range 2. To make the conditional probability for boundary variables depend on E, there must be more than two stochastic variables for each boundary. One way is using more stochastic variables like $p_L(x_{-1}x_0|x_1)$ and the other way is increasing the variables of conditions like $p_L(x_0|x_1x_2)$. In either way, there is another problem. The conserved density

can take three values 0, 1 and 2. If we want to study general cases, two parameters per boundary are necessary. Thus, the boundary condition can be more complicated than Eqs. (13) or (15). However, we can avoid these problems. Carefully looking at the time evolution of rule 56, one can notice that block 111 does not appear in time $t \geq 1$. In fact, no preimages for block 111 exist for rule 56. This means that $E(xyz) = x + y + z - 3xyz$ is equivalent to $E(xyz) = x + y + z$ for $t \geq 1$. Thus, rule 56 conserves the number of 1s for $t \geq 1$. Such a quantity was called an eventually conserved quantity in (Hattori and Takesue 1991). In the case of stochastic boundary condition, block 111 can appear only at the ends of the system. No 111 appears in the interior of the system. Therefore, we can use the boundary condition (13) for rule 56. The rule function $f_{56} = x(1-y) + (1-x)yz$ satisfies the following equation,

$$f_{56}(xyz) - y = x(1-y) - y(1-z) - xyz. \tag{18}$$

This is interpreted as that the current function for the eventually conserved quantity is $J(xy) = x(1-y)$.
Rule 11 has an eventually conserved quantity, too. In this case, block 101 has no preimages and under the absence of 101, $E(xy) = (x-y)^2$ becomes a conserved density. The corresponding current function is $J(xyz) = x(1-y) + (1-x)yz$. Therefore, the boundary condition (15) is appropriate to this rule.

5. Application of the domain wall theory

The domain wall theory is applied to the ECA with the stochastic boundary condition as follows. First we try to obtain the probability distribution of patterns of size three in the stationary states. Let p_i denote the probability distribution of block pattern starting from cell i. For example, $p_i(000)$ means that the probability that $x_i x_{i+1} x_{i+2} = 000$ and $p_i(0101)$ is the probability that $x_i x_{i+1} x_{i+2} x_{i+3} = 0101$. In the stationary state, those probabilities satisfy the following equations

$$p_i(x_1x_2x_3) = \sum_{x'_0,x'_1,x'_2,x'_3,x'_4} \delta(x_1x_2x_3, f(x'_0x'_1x'_2)f(x'_1x'_2x'_3)f(x'_2x'_3x'_4))p_{i-1}(x'_0x'_1x'_2x'_3x'_4). \tag{19}$$

Assuming uniformity for p_i and introducing decoupling approximation for the probability distribution of larger size if necessary, we can obtain a set of stationary solutions which contain the average density of the additive conserved quantity as a parameter. In particular, it is useful to use the logic that if $p(x_0x_1x_2x_3) = 0$, we must have $p(x_0x_1x_2) = 0$ or $p(x_1x_2x_3) = 0$ to close the equations consistently. Next, we try to adapt the solution to the left and right boundary conditions. If the boundary condition (15) is employed, the following equations must hold at the left boundary, namely

$$p_1(x_1x_2x_3) = \sum_{x'_0,x'_1,x'_2,x'_3,x'_4} \delta(x_1x_2x_3, f(x'_0x'_1x'_2)f(x'_1x'_2x'_3)f(x'_2x'_3x'_4))p_0(x'_0x'_1x'_2x'_3x'_4) \tag{20}$$

and

$$p_0(x_0x_1x_2x_3x_4) = p_L(x_0|x_1)p_1(x_1x_2x_3x_4). \tag{21}$$

By solving the equations, the distribution in the left phase is determined and the average density ρ_L and current J_L of the additive conserved quantity are obtained as functions of parameter α. The right phase is determined in the similar manner and the average density ρ_R and the average current J_R is computed as functions of β. Once these quantities are obtained,

we can apply the domain wall theory and the line of phase transition is calculated from the condition $J_L = J_R$.
In (Takesue 2008) we showed how the above procedure works for rule 184. In that case, use of the probability distributions of size two is sufficient. In the following, we will discuss the four rules 35, 14, 56 and 11 in this order.

5.1 Rule 35

If we assume uniformity, Eq. (19) becomes

$$\begin{aligned}
&p(000) = p(1100) + p(111), && p(001) = p(100) + p(1101),\\
&p(010) = p(101) + p(1001), && p(011) = p(1000),\\
&p(100) = p(011) + p(0100), && p(101) = p(0101),\\
&p(110) = p(0001), && p(111) = p(0000).
\end{aligned} \tag{22}$$

The sixth equation means $p(1101) = 0$, so we must have $p(110) = 0$ or $p(101) = 0$. In the former case, the first and second equations mean that $p(000) = p(111)$ and $p(001) = p(100)$. Then, $p(0001) = p(1000) = 0$ is obtained from the eighth equation. Therefore, we have $p(000) = p(011) = p(111) = 0$ and the remainings are determined as

$$p(001) = p(100) = 1 - \rho, \quad p(010) = \frac{\rho}{2}, \quad p(101) = \frac{3}{2}\rho - 1, \tag{23}$$

where ρ is the expectation value of $E(xy)$ and this solution makes sense only when $\frac{2}{3} \leq \rho \leq 1$. In the latter case, using approximation $p(0000) = \frac{p(000)^2}{p(00)}$, we arrive at

$$\begin{aligned}
&p(000) = \frac{2-3\rho}{4}, && p(001) = \frac{\rho}{2}, && p(010) = \frac{\rho^2}{2-\rho} && p(011) = \frac{\rho(2-3\rho)}{2(2-\rho)},\\
&p(100) = \frac{\rho}{2}, && p(101) = 0, && p(110) = \frac{\rho(2-3\rho)}{2(2-\rho)}, && p(111) = \frac{(2-3\rho)^2}{4(2-\rho)}.
\end{aligned} \tag{24}$$

This is the solution for $0 \leq \rho \leq \frac{2}{3}$.
The connection condition (20) is written as

$$\begin{aligned}
&p_1(000) = p_1(1100) + p_1(111), && p_1(001) = p_1(100) + p_1(1101),\\
&p_1(010) = p_1(101) + p_0(1001), && p_1(011) = p_0(1000),\\
&p_1(100) = p_1(011) + p_1(0100), && p_1(101) = p_1(0101),\\
&p_1(110) = p_0(0001), && p_1(111) = p_0(0000).
\end{aligned} \tag{25}$$

As in Eq. (21), p_0 is written as $p_0(xyzw) = (\alpha\delta_{x,1-y} + (1-\alpha)\delta_{xy})p_1(yzw)$. Then substitution of the solution for $0 \leq \rho \leq \frac{2}{3}$, (23), into p_1 satisfies the above equations if $\rho = \rho_L = \frac{2\alpha}{2+\alpha}$. Notice that as α varies from 0 to 1, ρ_L varies from 0 to $\frac{2}{3}$. The stationary solution for $\rho \geq \frac{2}{3}$, (24), cannot satisfy the above equations. At the right boundary $p_{N-2}(x_{N-2}x_{N-1}x_N)$ must satisfy

$$\begin{aligned}
&p_{N-2}(000) = p_{N-2}(1100) + p_{N-2}(111), && p_{N-2}(001) = p_{N-2}(100) + p_{N-2}(1101),\\
&p_{N-2}(010) = p_{N-2}(101) + p_{N-3}(1001), && p_{N-2}(011) = p_{N-3}(1000),\\
&p_{N-2}(100) = p_{N-2}(011) + p_{N-2}(0100), && p_{N-2}(101) = p_{N-2}(0101),\\
&p_{N-2}(110) = p_{N-3}(0001), && p_{N-2}(111) = p_{N-3}(0000).
\end{aligned} \tag{26}$$

Here the boundary condition is $p_{N-2}(xyzw) = [\beta\delta_{zw} + (1-\beta)\delta_{z,1-w}]p_{N-2}(xyz)$. In this case substitution of (24) into p_{N-2} satisfies the equations if $\rho = \rho_R = \frac{2}{2+\beta}$. Because $J(xyz) = x(1-y)(1-z)$, the flux in the left phase is $J_L = p_1(100) = \frac{\alpha}{2+\alpha}$ and that in the right phase is $J_R = p_{N-2}(100) = \frac{\beta}{2+\beta}$. Then, the velocity of the domain wall is obtained as

$$V = \frac{J_R - J_L}{\rho_R - \rho_L} = \frac{\beta - \alpha}{2 - \alpha - \alpha\beta} \tag{27}$$

Thus, if $\beta > \alpha$ the left phase prevails, and if $\beta < \alpha$ the right phase does.

5.2 Rule 14

Equations for a uniform stationary state are

$$\begin{aligned}
&p(000) = p(111) + p(0000) + p(11000), && p(001) = p(0001) + p(1101) + p(11001),\\
&p(010) = p(101), && p(011) = p(001),\\
&p(100) = p(011) + p(01000), && p(101) = p(0101) + p(01001),\\
&p(110) = p(001), && p(111) = 0.
\end{aligned} \tag{28}$$

The first and the eighth equations imply that $p(01000) = 0$. Thus, $p(010) = 0$ or $p(100) = 0$ or $p(000) = 0$ must be satisfied to obtain a consistent solution. Case $p(010) = 0$ leads to

$$p(010) = p(101) = p(111) = 0,\ p(001) = p(011) = p(100) = p(110) = \frac{\rho}{2},\ p(000) = 1 - 2\rho, \tag{29}$$

which is the solution for $0 \le \rho \le \frac{1}{2}$. Case $p(100) = 0$ leads to the solution $p(101) = p(010) = 2\rho$, $p(000) = 1 - 2\rho$ and the other entries are zeroes. In this case, block 000 and the other two cannot coexist in a system, because if 000 coexists with some 1s, 001 and 100 should have nonzero probability. Thus, this solution is not ergodic in the sense that time average for a system does not agree with expectation with respect to this probability distribution. Because we are interested in ergodic distribution only, we do not adopt this solution. In Case $p(000) = 0$ we have the following solution

$$p(000) = p(111) = 0,\ p(001) = p(011) = p(100) = p(110) = \frac{1-\rho}{2},\ p(010) = p(101) = \rho - \frac{1}{2} \tag{30}$$

It corresponds to case $\frac{1}{2} \le \rho \le 1$. The connection condition at the right end is given as

$$\begin{aligned}
&p_{N-2}(000) = p_{N-3}(111) + \beta[p_{N-2}(000) + p_{N-3}(1100)],\\
&p_{N-2}(001) = p_{N-3}(1101) + (1-\beta)[p_{N-2}(000) + p_{N-3}(1100)],\\
&p_{N-2}(010) = p_{N-3}(101), && p_{N-2}(011) = p_{N-2}(001),\\
&p_{N-2}(100) = p_{N-3}(100) + \beta p_{N-3}(0100),\\
&p_{N-2}(101) = p_{N-3}(0101) + (1-\beta)p_{N-3}(0100),\\
&p_{N-2}(110) = p_{N-3}(001), && p_{N-2}(111) = 0.
\end{aligned} \tag{31}$$

Substitution of the stationary solution for $0 \le \rho \le \frac{1}{2}$, (29), into p_{N-2} satisfies these equations and the density of the conserved quantity is obtained as $\rho = \rho_R = \frac{2(1-\beta)}{4-3\beta}$. In addition, the average current is given as

$$J_R = -\rho_R = -\frac{2(1-\beta)}{4-3\beta}. \tag{32}$$

The connection condition at the left end is given as

$$
\begin{aligned}
&p_1(000) = (1-\alpha)(p_1(111)+p_1(1000))+p_1(0000), && \\
&p_1(001) = p_1(0001)+(1-\alpha)(p_1(101)+p_1(1001)), && \\
&p_1(010) = \alpha p_1(01), && p_1(011) = p_1(001), \\
&p_1(100) = \alpha(p_1(11)+p_1(1000)), && p_1(101) = \alpha(p_1(101)+p_1(1001)), \\
&p_1(110) = (1-\alpha)p_1(01), && p_1(111) = 0.
\end{aligned} \tag{33}
$$

In this case, if either of the uniform stationary solutions is inserted into p_1, the equations are not satisfied. Instead, if we assume that p_2 equals the stationary solution for $\frac{1}{2} \le \rho \le 1$, we have

$$
\begin{aligned}
p_1(000) &= \frac{(1-\alpha)^2}{4-3\alpha+\alpha^2}, & p_1(001) &= \frac{1-\alpha}{4-3\alpha+\alpha^2}, \\
p_1(010) &= \frac{\alpha}{4-3\alpha+\alpha^2}, & p_1(011) &= \frac{1-\alpha}{4-3\alpha+\alpha^2}, \\
p_1(100) &= \frac{\alpha(1-\alpha)}{4-3\alpha+\alpha^2}, & p_1(101) &= \frac{\alpha^2}{4-3\alpha+\alpha^2}, \\
p_1(110) &= \frac{1-\alpha}{4-3\alpha+\alpha^2}, & p_1(111) &= 0.
\end{aligned} \tag{34}
$$

Note that only the property $p_2(000) = 0$ has been used to derive the above. The relation between α and ρ should be obtained by $p_1(000)+p_1(100) = p_2(000)+p_2(001) = \frac{1-\rho}{2}$, which leads to

$$
\rho = \rho_L = \frac{2-\alpha+\alpha^2}{4-3\alpha+\alpha^2}. \tag{35}
$$

However, this is inconsistent with another condition $p_1(001)+p_1(101) = p_2(010)+p_2(011) = \frac{\rho}{2}$. Namely, the two conditions cannot be satisfied at the same time by the uniform solution. This inconsistency is resolved by considering a periodic solution for the stationary distribution. If we assume period two for the distribution and denote the distribution function starting from an even-numbered cell by p_{e} and that starting from an odd-numbered cell by p_{o}, Eqs. 28 are replaced with

$$
\begin{aligned}
&p_{\mathrm{e}}(000) = p_{\mathrm{o}}(111)+p_{\mathrm{e}}(0000)+p_{\mathrm{o}}(10000), && p_{\mathrm{e}}(001) = p_{\mathrm{e}}(0001)+p_{\mathrm{o}}(1101)+p_{\mathrm{o}}(11001), \\
&p_{\mathrm{e}}(010) = p_{\mathrm{o}}(101), && p_{\mathrm{e}}(011) = p_{\mathrm{e}}(001), \\
&p_{\mathrm{e}}(100) = p_{\mathrm{o}}(011)+p_{\mathrm{o}}(01000), && p_{\mathrm{e}}(101) = p_{\mathrm{o}}(0101)+p_{\mathrm{o}}(01001), \\
&p_{\mathrm{e}}(110) = p_{\mathrm{o}}(001) && p_{\mathrm{e}}(111) = 0
\end{aligned} \tag{36}
$$

and those with p_{e} and p_{o} interchanged. For $\rho \ge \frac{1}{2}$, we have the solution similar to (30) except

$$
p_{\mathrm{e}}(010) = p_{\mathrm{o}}(101) = \rho - \frac{1}{2} + \epsilon, \quad p_{\mathrm{e}}(101) = p_{\mathrm{o}}(010) = \rho - \frac{1}{2} - \epsilon, \tag{37}
$$

where ϵ is a parameter satisfying $-(\rho - 1/2) \le \epsilon \le \rho - 1/2$. Then, if we assume $p_2 = p_{\mathrm{e}}$ with

$$
\epsilon = -\frac{\alpha(1-\alpha)}{2(4-3\alpha+\alpha^2)}, \tag{38}
$$

the solution is consistently connected to Eq. 34. Thus, the average density (35) is still correct and the average current in the left phase is given by

$$J_L = -p_{\rm e}(001) - p_{\rm e}(110) - p_{\rm e}(111) = -\frac{2(1-\alpha)}{4-3\alpha+\alpha^2} \tag{39}$$

The phase transition line is calculated via $J_L = J_R$ and the result is

$$\beta = \frac{\alpha(1+\alpha)}{1+\alpha^2}. \tag{40}$$

5.3 Rule 56

Equations for a uniform stationary state are

$$\begin{aligned}
&p(000) = p(0000) + p(1111) + p(00010), && p(001) = p(0010) + p(1110) + p(00011),\\
&p(010) = p(0011) + p(100) + p(1010), && p(011) = p(1011),\\
&p(100) = p(1000) + p(0111) + p(10010), && p(101) = p(0110) + p(1010) + p(10011),\\
&p(110) = p(1011), && p(111) = 0.
\end{aligned} \tag{41}$$

The first and the eighth equations leads to $p(00011) = 0$ and the second equation means $p(10011) = 0$. Thus $p(0011) = 0$ must hold. This implies that $p(001) = 0$ or $p(011) = 0$. In the former case, we have $p(001) = p(100) = p(111) = 0$. Then, for the state to be ergodic $p(000)$ must vanish and the remainings are determined as

$$p(010) = 2-3\rho, \quad p(011) = p(110) = 2\rho - 1, \quad p(101) = 1-\rho, \tag{42}$$

where ρ is the density of 1s. This solution makes sense when $\frac{1}{2} \le \rho \le \frac{2}{3}$. Note that $p(111) = 0$ means the maximum value of ρ is $\frac{2}{3}$. In the latter case, $p(011) = p(110) = p(111) = 0$ and the others are

$$p(000) = 1-2\rho-a, \quad p(001) = p(100) = a, \quad p(010) = \rho, \quad p(101) = \rho - a, \tag{43}$$

where a is a real number in the region $0 \le a \le \min(\rho, 1-2\rho)$. This is the solution for $0 \le \rho \le \frac{1}{2}$. The connection at the left end is given by

$$\begin{aligned}
&p_1(000) = (1-\alpha)[p_1(000) + p_1(0010)],\\
&p_1(001) = (1-\alpha)[p_1(010) + p_1(0011)] + \alpha p_1(110),\\
&p_1(010) = (1-\alpha)p_1(011) + p_1(100) + p_1(1010), && p_1(011) = p_1(1011),\\
&p_1(100) = \alpha[p_1(000) + p_1(0010)],\\
&p_1(101) = \alpha[p_1(010) + p_1(0011)] + (1-\alpha)p_1(110),\\
&p_1(110) = \alpha p_1(011), && p_1(111) = 0.
\end{aligned} \tag{44}$$

This is satisfied by the bulk solution for $0 \le \rho \le \frac{1}{2}$ with $\rho = \rho_L = \frac{\alpha}{1+\alpha}$ and $a = \frac{\alpha(1-\alpha)}{1+\alpha}$. The average current in this phase is

$$J_L = p_1(100) + p_1(101) = \rho_L = \frac{\alpha}{1+\alpha}. \tag{45}$$

The connection at the right end is given as

$$
\begin{aligned}
&p_{N-2}(000) = p_{N-3}(0000) + \beta p_{N-3}(0001),\\
&p_{N-2}(001) = p_{N-3}(0010) + (1-\beta) p_{N-3}(0001),\\
&p_{N-2}(010) = p_{N-2}(100) + p_{N-3}(0011) + \beta p_{N-2}(101), \qquad && p_{N-2}(011) = p_{N-2}(1011),\\
&p_{N-2}(100) = p_{N-3}(1000) + \beta p_{N-3}(1001),\\
&p_{N-2}(101) = p_{N-3}(0110) + p_{N-3}(1010) + (1-\beta) p_{N-3}(1001),\\
&p_{N-2}(110) = p_{N-3}(1011), && p_{N-2}(111) = 0.
\end{aligned} \tag{46}
$$

This is satisfied by substitution of the stationary solution with $\rho = \rho_R = \frac{2-\beta}{3-\beta}$ into p_{N-2}. The average current is given as

$$
J_R = p_{N-2}(100) + p_{N-2}(101) = 1 - \rho_R = \frac{1}{3-\beta} \tag{47}
$$

The phase transition occurs when $J_L = J_R$, which is given by

$$
\beta = 2 - \frac{1}{\alpha}. \tag{48}
$$

5.4 Rule 11

Let us assume period-two stationary solutions from the beginning. The equations are

$$
\begin{aligned}
&p_{\mathrm{e}}(000) = p_{\mathrm{o}}(111) + p_{\mathrm{e}}(1010) + p_{\mathrm{o}}(10100), && p_{\mathrm{e}}(001) = p_{\mathrm{e}}(100) + p_{\mathrm{e}}(1011),\\
&p_{\mathrm{e}}(010) = p_{\mathrm{o}}(1011) + p_{\mathrm{o}}(10010), && p_{\mathrm{e}}(011) = p_{\mathrm{o}}(1000) + p_{\mathrm{o}}(10011),\\
&p_{\mathrm{e}}(100) = p_{\mathrm{o}}(011) + p_{\mathrm{o}}(0010), && p_{\mathrm{e}}(101) = 0,\\
&p_{\mathrm{e}}(110) = p_{\mathrm{o}}(0011) + p_{\mathrm{o}}(00010) && p_{\mathrm{e}}(111) = p_{\mathrm{o}}(0000) + p_{\mathrm{o}}(00010).
\end{aligned} \tag{49}
$$

and those with p_{e} and p_{o} interchanged. The sixth equation implies the second and third terms in the rhs of the first equation vanish. Thus we obtain $p_{\mathrm{e}}(000) = p_{\mathrm{o}}(111)$. In the same manner, $p_{\mathrm{e}}(111) = p_{\mathrm{o}}(000)$, $p_{\mathrm{e}}(001) = p_{\mathrm{e}}(100)$ and $p_{\mathrm{o}}(001) = p_{\mathrm{o}}(100)$ are obtained. The eighth equation is rewritten as $p_{\mathrm{e}}(111) = p_{\mathrm{o}}(000) - p_{\mathrm{o}}(00010)$ and substitution of $p_{\mathrm{e}}(111) = p_{\mathrm{o}}(000)$ leads to $p_{\mathrm{o}}(00010) = 0$. Thus at least one of $p_{\mathrm{o}}(000)$, $p_{\mathrm{e}}(001)$, or $p_{\mathrm{o}}(010)$ must vanish. Similarly, one of $p_{\mathrm{e}}(000)$, $p_{\mathrm{o}}(001)$, or $p_{\mathrm{e}}(010)$ must vanish. However, if $p_{\mathrm{e}}(001) = 0$, $p_{\mathrm{o}}(010)$ must vanish, because $p_{\mathrm{o}}(01) = p_{\mathrm{e}}(101) + p_{\mathrm{e}}(001) = 0$. Thus, the case $p_{\mathrm{e}}(001) = 0$ is included in the case $p_{\mathrm{o}}(010) = 0$. Moreover, if $p_{\mathrm{e}}(010) = 0$, $p_{\mathrm{o}}(10010) = 0$ and accordingly $p_{\mathrm{e}}(0010) = p_{\mathrm{o}}(010) = 0$. Thus, it is sufficient to investigate the two cases $p_{\mathrm{e}}(000) = p_{\mathrm{o}}(000) = 0$ and $p_{\mathrm{e}}(010) = p_{\mathrm{o}}(010) = 0$. The former case leads to the following uniform (namely $p_{\mathrm{e}} = p_{\mathrm{o}} = p$) solution

$$
\begin{aligned}
&p(000) = p(101) = p(111) = 0, && p(001) = p(100) = \frac{\rho}{2},\\
&p(010) = 2\rho - 1, && p(011) = p(110) = 1 - \frac{3}{2}\rho,
\end{aligned} \tag{50}
$$

where the average density ρ of eventually conserved quantity satisfies $\frac{1}{2} \leq \rho \leq \frac{2}{3}$. In the latter case, we have the period-two solution as

$$\begin{aligned}
&p_{\rm e}(000) = \frac{1}{2} - \rho + \epsilon, && p_{\rm o}(000) = \frac{1}{2} - \rho - \epsilon, \\
&p_{\rm e}(001) = p_{\rm e}(100) = \frac{\rho}{2} - \epsilon, && p_{\rm o}(001) = p_{\rm o}(100) = \frac{\rho}{2} + \epsilon, \\
&p_{\rm e}(010) = p_{\rm e}(101) = 0, && p_{\rm o}(010) = p_{\rm o}(101) = 0, \\
&p_{\rm e}(011) = p_{\rm e}(110) = \frac{\rho}{2} + \epsilon, && p_{\rm o}(011) = p_{\rm o}(110) = \frac{\rho}{2} - \epsilon, \\
&p_{\rm e}(111) = \frac{1}{2} - \rho - \epsilon, && p_{\rm o}(111) = \frac{1}{2} - \rho + \epsilon,
\end{aligned} \tag{51}$$

where $0 \leq \rho \leq \frac{1}{2}$ and $0 \leq \epsilon \leq \min(\rho/2, 1/2 - \rho)$. Connection at the left end is given by

$$\begin{aligned}
&p_1(000) = (1-\alpha)p_1(11) + p_1(1010) + \alpha p_1(0100), && p_1(001) = p_1(100), \\
&p_1(010) = \alpha[p_1(011) + p_1(0010)], && p_1(011) = \alpha[p_1(000) + p_1(0011)], \\
&p_1(100) = \alpha p_1(11) + (1-\alpha)p_1(010), && p_1(101) = 0, \\
&p_1(110) = (1-\alpha)[p_1(011) + p_1(0010)], && p_1(111) = (1-\alpha)[p_1(000) + p_1(0011)].
\end{aligned} \tag{52}$$

These equations can be solved with the assumption that p_2 is given by $p_{\rm e}$ in Eq. (51). As the result, p_1 is obtained as

$$\begin{aligned}
&p_1(000) = K\left(1 - \alpha - \alpha^2 + 2\alpha^3\right), && p_1(001) = K\alpha\left(1 + \alpha - 2\alpha^2\right), \\
&p_1(010) = K\alpha^2, && p_1(011) = K\alpha, \\
&p_1(100) = K\alpha\left(1 + \alpha - 2\alpha^2\right), && p_1(101) = 0, \\
&p_1(110) = K\alpha(1-\alpha), && p_1(111) = K(1-\alpha),
\end{aligned} \tag{53}$$

where $K = (2 + 2\alpha + \alpha^2 - 2\alpha^3)^{-1}$. The relations among ρ_L, ϵ and α are obtained via the consistency between p_1 and p_2 as

$$\rho_L = 1 - 2K = \frac{\alpha(2 + \alpha - 2\alpha^2)}{2 + 2\alpha + \alpha^2 - 2\alpha^3}, \quad \epsilon = \frac{\alpha^2(1-\alpha)}{2 + 2\alpha + \alpha^2 - 2\alpha^3}. \tag{54}$$

The conserved current in the left domain is $J_L = p_{\rm e}(011) + p_{\rm e}(100) = \rho_L = K\alpha(2 + \alpha - 2\alpha^2)$. Connection at the right end is given by the condition

$$\begin{aligned}
&p_{N-2}(000) = p_{N-3}(000) + \beta p_{N-3}(0001), && p_{N-2}(001) = p_{N-2}(100) \\
&p_{N-2}(010) = (1-\beta)p_{N-3}(1001), && p_{N-2}(011) = p_{N-3}(1000) + \beta p_{N-3}(1001), \\
&p_{N-2}(100) = p_{N-3}(011) + p_{N-3}(0010), && p_{N-2}(101) = 0 \\
&p_{N-2}(110) = p_{N-3}(0011) + (1-\beta)p_{N-3}(0001), && p_{N-2}(111) = p_{N-3}(0000) + \beta p_{N-3}(0001).
\end{aligned} \tag{55}$$

These equations are satisfied by the substitution of (50) into p_{N-2}. Relation between ρ_R and β is given by

$$\rho_R = \frac{2}{3 + \beta}. \tag{56}$$

And the average current in the right phase is given as $J_R = 1 - \rho_R = \frac{1+\beta}{3+\beta}$. The phase transition occurs when $J_L = J_R$, that is

$$\beta = (2\alpha - 1)(1 - \alpha^2). \tag{57}$$

5.5 Numerical simulations

We carried out simulations of the four rules for various α and β. The result is illustrated in Fig. 4. As is seen from the figure, the agreement of the theoretically obtained phase transition line and numerical results is excellent. Values of the density and the current also show a nice agreement between the theory and numerical results.

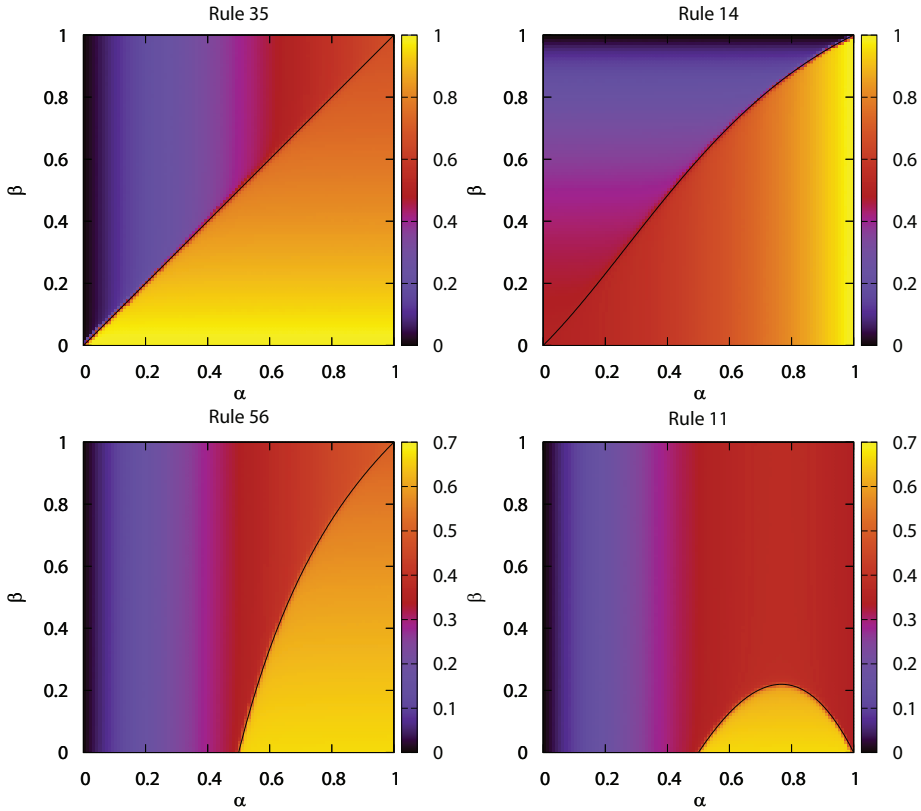


Fig. 4. Phase diagrams of the ECAs. The density of the conserved quantity for each rule and the theoretically obtained phase transition lines are illustrated.

6. Diffusion of the domain wall

The velocity of the domain wall V vanishes on the phase transition line. It does not mean that the domain wall stops somewhere but implies diffusive motion. Eq. (11) can be interpreted as that the domain wall undergoes a random walk with the rate of hopping to the right is

$J_R/(\rho_R - \rho_L)$ and that to the left is $J_L/(\rho_R - \rho_L)$. Then the diffusion constant is given by

$$D = \frac{J_R + J_L}{2(\rho_R - \rho_L)}. \tag{58}$$

On the phase transition line, the drift velocity vanishes and pure diffusive motion appears. This diffusive motion leads to interesting power-law in the power spectral density for the time sequence of the conserved density at a position (Takesue et al 2003). This is first observed for the TASEP and our ECA rules share this phenomenon. Now we consider the case where the conserved density is $E(xy) = (x - y)^2$. The case of $E(x) = x$ is simpler than that. The power spectral density is defined as follows. Record a time sequence of $E(x_i^t x_{i+1}^t)$ for a fixed i and $0 \leq t < T$. Fourier components of the time sequence are calculated as

$$\phi_n = \frac{1}{T} \sum_{t=0}^{T-1} \left(E(x_i^t x_{i+1}^t) - \rho_i \right) e^{-i\omega_n t}, \tag{59}$$

where $\omega_n = \frac{2\pi n}{T}$ $(n = 0, 1, 2, \ldots, T)$ and ρ_i is the expectation value of $E(x_i x_{i+1})$ in the stationary state. Then, the power spectral density is defined as

$$I(\omega_n) = T\langle |\phi_n|^2 \rangle, \tag{60}$$

where $\langle\ \rangle$ denotes the sample average. Now, we approximate the motion of the domain wall by Brownian motion. In this approximation, the density profile is represented as

$$\rho(x, t) = \rho_L + (\rho_R - \rho_L)\theta(x - X(t)), \tag{61}$$

where $\rho(x, t)$ is the coarse-grained density profile at time t and position x, $X(t)$ is a Brownian motion with diffusion constant D, $\theta(x)$ is the Heaviside function. Applying the Wiener-Khinchin theorem, we obtain the power spectral density as

$$I(\omega) = \frac{\sqrt{2D}}{2N} (\rho_R - \rho_L)^2 \omega^{-3/2}. \tag{62}$$

The power spectral density for rule 35 is shown in Fig. 5. The power law behavior is seen in a region of ω. The agreement with the theory is very good including the prefactor. Also in the other rules, the power spectral density exhibits the power law on the phase transition line. The prefactor agrees with the theoretical value in every case except in rule 11, where a small deviation is seen. The deviation is expected to decrease as the system size increases.

7. Discussion and conclusion

We have seen that the ECA with a single additive conserved quantity show a nonequilibrium phase transition of the same type as in the ASEP. The phase transition line is precisely determined by applying the domain wall theory. In these rules, there occurs only the first-order phase transition and no maximal current phase is observed. It is to be investigated why the maximal current phase does not exist in the ECA and in what kind of CA it appear. The connection between the stationary distribution and the boundary condition depends on the conditional probability. We discuss some possible generalizations. For example, we can consider the following conditional probability.

$$p_L(1|0) = \alpha, \quad p_L(0|0) = 1 - \alpha, \quad p_L(0|1) = \gamma, \quad p_L(1|1) = 1 - \gamma, \tag{63}$$

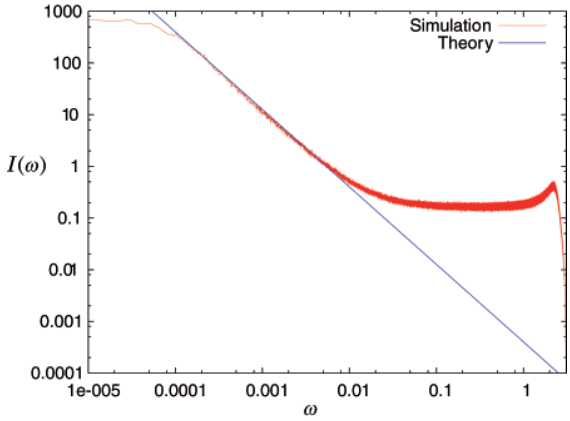


Fig. 5. Power spectral density for rule 35 with $\alpha = \beta = 0.5$. The system size $N = 200$ and the simulation time $T = 2^{20}$.

which is the general form of $p_L(x|y)$. In rule 11, this cause a small modification of the connected stationary state and the resulting phase transition line is represented as $\beta = (1-\alpha^2)(\alpha+\gamma-1)$. The similar generalization of the right boundary

$$p_R(0|0) = \delta, \quad p_R(1|0) = 1-\delta, \quad p_R(0|1) = 1-\beta, \quad p_R(1|1) = \beta \tag{64}$$

does not affect the connection.
In rules 11 and 14, the stationary distribution connecting with the left boundary is periodic with period two, which was numerically confirmed. This is caused from the fact that p_1 cannot satisfy a property of the stationary distribution, that is $p(010) = 0$ for rule 11 and $p(000) = 0$ for rule 14. However, this is resolved if we employ a suitable conditional probability of the form $p(x_0|x_1x_2)$. For rule 11, if we use the conditional probability given as

$$p'_L(1|01) = 0, \quad p'_L(0|01) = 1, \quad \text{and otherwise } p'_L(x|yz) = p_L(x|y), \tag{65}$$

where p_L is that defined in Eq. (16), $p_1(010) = 0$ is realized and p_1 agrees with the uniform solution of the stationary state ($\epsilon = 0$). The pase transition line is drastically changed into $\beta = 2\alpha - 1$. In the same manner, utilizing the conditional probability

$$p'_L(1|11) = 0, \quad p'_L(0|11) = 1, \quad \text{and otherwise } p'_L(x|yz) = p_L(x|y) \tag{66}$$

for rule 14, p_1 connects with the uniform stationary solution and the phase transition occurs on the line $\beta = \frac{2\alpha}{1+\alpha}$.

8. References

[Capcarrere 2001] M.S. Capcarrère and M. Sipper. (2001). *Necessary conditions for density classification by cellular automata*, *Physical Review Letters*. 64, 036113.

[Creutz 1985] M. Creutz, (1985). *Deterministic Ising Dynamics*, *Annals of Physics*, 167, 62 – 72.

[Derrida et al. 1993] B. Derrida, M.R. Evans, V. Hakim and V. Pasquier. (1993). *Exact solution of a 1D asymmetric exclusion model using a matrix formulation*, *Journal of Physics A: Mathematical and General*, 26, 1493 – 1517.

[Hattori and Takesue 1991] T. Hattori & S. Takesue, (1991). *Additive Conserved Quantities in Discrete-Time Lattice Dynamical Systems*, *Physica D*, 49, 295 – 322.

[Kolomeisky et al. 1998] A.B. Kolomeisky, G.M. Schütz, E.B. Kolomeisky, and J.P. Straley. (1998). *Phase diagram of one-dimensional driven lattice gases with open boundaries*, *J. Phys. A: Math. Gen.*, 31, 6911 – 6919.

[Land and Belew 1995] M. Land and R.K. Belew. (1995). *No Perfect Two-State Cellular Automata for Density Classification Exists*, *Physical Review Letters*, 74, 5148 – 5150.

[Saito et al. 1999] K.Saito, S. Takesue, and S. Miyashita, (1999). *Transport Phenomena at a Critical Point: Thermal Conduction in the Creutz Cellular Automaton*, *Physical Review E*, 59, 2783 – 2794.

[Sasamoto 1999] T. Sasamoto, (1999). *One-dimensional partially asymmetric simple exclusion process with open boundaries: Orthogonal polynomials approach*, *Journal of Physics A: Mathmatical and General*, 32, 7109 – 7131.

[Takesue 1987] S. Takesue, (1987). *Reversible Cellular Automata and Statistical Mechanics*, *Physical Review Letters*, 59, 2499 – 2502.

[Takesue 1989] S. Takesue, (1989). *Ergodic Properties and Thermodynamic Behavior of Elementary Reversible Cellular Automata. I. Basic Properties*, *Journal of Statistical Physics*, 56, 371 – 402.

[Takesue 1990a] S. Takesue, (1990). *Fourier's Law and the Green-Kubo Formula in a Cellular Automaton Model*, *Physical Review Letters*, 64, 252 – 255.

[Takesue 1990b] S. Takesue, (1990). *Relaxation Properties of Elementary Reversible Cellular Automata*, *Physica D*, 45, 278 – 284.

[Takesue 1995] S. Takesue, (1995). *Staggered Invariants in Cellular Automata*, *Complex Systems*, 9, 149 – 168.

[Takesue 1997] S. Takesue, (1997). *Boltzmann-Type Equations for Elementary Reversible Cellular Automata*, *Physica D*, 103, 190 – 200.

[Takesue et al 2003] S. Takesue, T. Mitsudo, and H. Hayakawa. (2003). *Power-law behavior in the power spectrum induced by Brownian motion of a domain wall*, *Physical Review E*, 68, 015103(R).

[Takesue 2008] S. Takesue, (2008). *Characteristic Dynamics of Elementary Cellular Automata with a Single Conserved Quantity*, *Journal of Cellular Automata*, 3, 123 – 144.

[Uchiyama 2004] M. Uchiyama, T. Sasamoto, and M. Wadati, (2004). *Asymmetric simple exclusion process with open boundaries and Askey-Wilson polynomials*, *Journal of Physics A: Mathematical and General*, 37, 4985 – 5002.

[Vichniac 1984] Gérard Y. Vichniac, (1984). *Simulating Physics with Cellular Automata*, *Physica D*, 10, 96 – 116.

[Wolfram 1983] S. Wolfram, (1983). *Statistical mechanics of cellular automata*, *Reviews of Modern Physics*, 55, 601 – 644

20

Cellular Automata – a Tool for Disorder, Noise and Dissipation Investigations

W. Leoński[1] and A. Kowalewska-Kudłaszyk[2]
[1]*Quantum Optics and Engineering Division, Institute of Physics, University of Zielona Góra, Zielona Góra*
[2]*Department of Physics, Adam Mickiewicz University, Poznań Poland*

1. Introduction

There are many problems that are impossible or very difficult to solve with numerical methods. In particular, they are related to the systems containing many subsystems for instance, atoms, molecules or even water drops and sand grains. Moreover, to this group we can number many continuous systems that are treated as a discrete set of many sub-systems. Although the dynamics of many of them can be described by the set of differential equations, such attempts are futile. Due to the complexity of the whole system, finding solutions for such huge amount of equations is often impracticable. For these cases the cellular automata (CA) formalism seems to be one of possible solutions.

Although the idea of CA was proposed during II World War in the Los Alamos laboratory by Stanisław Ulam and Janos von Neumann, for its real development and practical applications we had to wait three decades (for references concerning those early papers concerning CA simulations see for instance Butler (1973) *and the references quoted therein*). In consequence, from the 1970-s we can record growing number of papers devoted to CA applications in various fields of physics, biology, medicine, sociology, economy and other. For instance, one should mention the paper by Margolus et al. (1986) who applied CA for the fluid dynamics modelling. CA formalism can also be used for Brownian motion investigation, for instance for the solid-fluid suspensions (Ladd & Colvin (1988)). For solid state systems CA formalism allows simulation of magnetisation processes (Vichniac (1984)). Moreover, CA investigation concerning laser dynamics was presented by Guisado et al. (2003), whereas N-body systems with multi-step potentials were discussed by Lejeune et al. (1999). It should be emphasized that CA formalism was applied not only for physical systems investigation but also for medical (for instance see Barclay et al. (2010)), sociological (Dabbaghian et al. (2010)), traffic flow (Combinido & Lim (2010)), agriculture (Zeng et al. (2010)), image processing (Rosin (2010)) and others. These are only some selected recent examples of the papers dealing with a very broad and diverse subject of CA applications.

CA can be defined as dynamical system that is discrete in time and space and is characterized by discrete states. This rather general description is far from the mathematical exactness and precision however, it reflects the basic properties of CA. Of course one should realize that there are various strict mathematical CA definitions. At this point we shall present such CA

definition that is not mathematically pure, but can be readily applied by the reader in his further investigation. So, the cellular automaton $\mathcal{A}$ can be defined as:

- n-dimensional lattice of cells $\mathcal{L}^n$,
- the finite set $\mathcal{S}$ of states of the cell,
- the neighbourhood of the cell $\mathcal{N} \subset \mathcal{L}^n$,
- the rules determining the state of given cell in the next time-step on the basis of its current state and the state of its neighbours.

In this paper we shall concentrate on practical realization of a such defined CA. We intend to show how such simple CA definitions allow construction of the model reflecting physical properties of the real system. Moreover, our aim was to present how a complicated system evolution can be investigated with the help of CA. In particular we shall discuss the model of many two-level subsystems. Two-level systems were used and discussed extensively in physical models of solid state physics or quantum optics (for discussion concerning the latter see the classical book by Allen & Eberly (2007)). Obviously such two-level systems can also be discussed as sociological or economical models. For instance, two-level 0-1 system may be related to the voting or *sell-buy* decisions. At this point it is worth mentioning the *Sznajd model* (Kondrat & Sznajd-Weron (2010) *and the references quoted therein*) describing physical Ising model and the sociological voting one as well. In general, one can say that many two-level system models can be applied to the energy or information dynamics simulations.

2. The model

In this chapter we shall concentrate on one-dimensional simple model comprising many two-level subsystems. This model was initially proposed by Walczak & Leoński (2003), where the first, preliminary considerations were presented. The Authors concentrated on the relations between the model proposed and a cavity with two-level atoms dynamics. They discussed the dependence of the system energy on the absorption probability for the individual atom. Moreover, for this model the entropic measures of disorder was defined and discussed in the papers Kowalewska-Kudłaszyk & Leoński (2006; 2008).

The model is organized as follows. First, we assume that we have a chain of two-level subsystems (atoms, cells). Each of them can be in one of the two states: ground or excited. These states will be denoted by 0 and 1, respectively. Obviously, depending on the application of our model, the states can be labelled for instance, as "up" and "down", "buy" and "sell" *etc.* Each of excited atoms can "emit energy (photon)" toward one of the two directions (left or right). If such emitted quanta of energy meet the atom in the ground state, it can absorb the energy. Since, our model has the probabilistic character, we assume some values for probabilities of the energy absorption and emission, and for the direction of the emission as well.

At the beginning of our numerical experiment we assume some length N of the system considered (number of cells) and the values of the emission and absorption probabilities: p_e and p_a, respectively. These probabilities are identical for all cells and remain constant during the whole process. Moreover, we assume which cells are initially excited. This initial cells states selection can be either done by some assumption (deterministic initial conditions) or performed randomly (random initial conditions).

Then we determine the simulation rules and thus, our simulation algorithm can be summarized in the following points:

- If for the time t particular atom $\{i\}$ is in its excited state, we take some random number $r_{it} \in < 0, 1 >$. If $r_{it} < p_e$ the atom in the next step (time $t + 1$) emits quantum of energy.
- If the energy is emitted, we take another random number $r_j \in < 0, 1 >$ – for $r_j < 1/2$ the quantum goes left (for $r_j \geq 1/2$ it goes right).
- If the photon meets the excited atom it passes this cell and goes farther (the atom is "transparent for radiation").
- If the quantum reaches the atom in its ground state we take the next random number $r_{jt} \in < 0, 1 >$. If $r_{jt} < p_a$ the atom in the next step (time $t + 1$) absorbs photon. For the opposite case the photon goes to the next cells.

The next point is determination of boundary conditions of the model discussed. Thus, we shall consider two cases. For the first of them (periodic boundary conditions) we assume that the first and the N^{th} cells are the same. It corresponds to the real systems of the *ring cavities* commonly discussed in optical models. For this case the total "energy" of the system is preserved, since we neglect all dissipation processes. Such "energy" can be calculated as a sum of all excited atoms at the given moment of time located within the system considered.
The other possibility is to assume that the energy can leak out of the system. For this case we can assume that at the ends of the cavity we have semi-transparent "mirrors". According to it we need to choose some reflection probability R at the beginning of the simulation. We assume that if the quantum of energy ("photon") reaches the "mirror" confining the cavity, we take a random number $r_R \in < 0, 1 >$. For the case when $r_R < R$ the photon remains inside the cavity and start to move in the opposite direction. If $r_R \geq R$ the quantum of energy disappears and the total system energy decreases.
One can see that our model has discrete character. Both time t and the position of the cell are discrete, which is characteristic of CA. Thanks to this fact, the parameters describing the system will change jumping from one value to another showing step-function character. Since we need continuously changing parameters we should avoid such behaviour. Therefore, we shall perform many simulations for the same values of the initial parameters and then, average the results. This problem was discussed for the case of the system energy in the earlier mentioned (Walczak & Leoński (2003)), where the first description of the model considered was presented.

3. Simulation results

At the beginning we present the exemplary simulation results for the system confined by "mirrors" with the reflection probability R assumed to be equal 0.25. We deal with a simple 13-cells model assuming that only three of them are initially excited. From Fig.1 one can see that for the time $t = 0$ we have excitations at the cell positions 9, 10 and 11. During the evolution, these excitations walk randomly right and left reaching the borders of the automaton in the time equal to 46 (right) and 51 (left). Since the reflection probability has a non-zero value, the "photons" are reflected and the system's energy is conserved. However, for $t = 58$ (right) and $t = 59$ (left) both energy quanta are emitted from the system. As a result from $t = 60$ we observe random walk of only one excitation.

3.1 Rate of energy leakage

At this point we shall concentrate on the CA model corresponding to the cavity with "mirrors" and we assume that the reflection probability R is smaller than unity. Since such a model can include dissipation effects, it allows the total energy leakage investigation. In particular, we

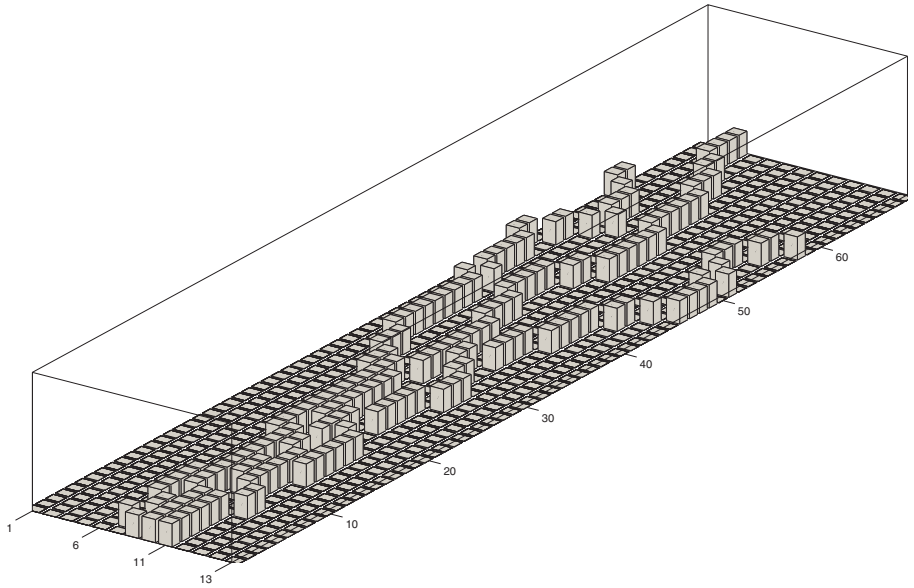


Fig. 1. A map visualising the spread of excitation from the three initially excited cells with time (from 1 to 70) for $p_a = 1.00$, $p_e = 0.25$ and the reflection probability $R = 0.25$. CA contains 13 cells.

shall be interested in the energy loss rate. Analysis of this rate behaviour can be a good test of whether our model reflects the properties of the real systems properly or not.
Therefore, at this point we define the rate related parameter $r = E_{n+1} - E_n$ as a differences between two subsequent total system energies. We do not divide such difference by the time, as it is equal to unity. This is a result of the discrete character of CA – within CA models time is discrete and numbered by the subsequent natural numbers.
It should be stressed out that to find the time-dependence of r we have to perform a series of simulation experiments for the same values of parameters describing the system. For a single simulation we obtain the energy time-dependence of the step-like function character. Therefore, to obtain a sufficiently smooth form of the results, we performed some number of simulations (for our case it was $\sim$ 100) and then, we averaged the results. Sometimes such averaging procedure was not sufficient and we needed additional smoothing procedure application. For the cases discussed here we applied LOESS (or LOWESS) – locally weighted scatterplot smoothing procedure (Cleveland (1979); Cleveland et al.& Devlin (1988)). This method allowed correct calculations of r without losing features appearing in the results.
At this point we shall concentrate on the deterministically chosen initial conditions. Thus, we assume that our model contains 150 cells. Those labelled by numbers 51–100 (located at the centre of the line) are excited, whereas the remaining ones are in their ground states. We assume that the absorption probability $p_a = 1.00$. Moreover, the probability of emission is assumed to be equal to 0.25.

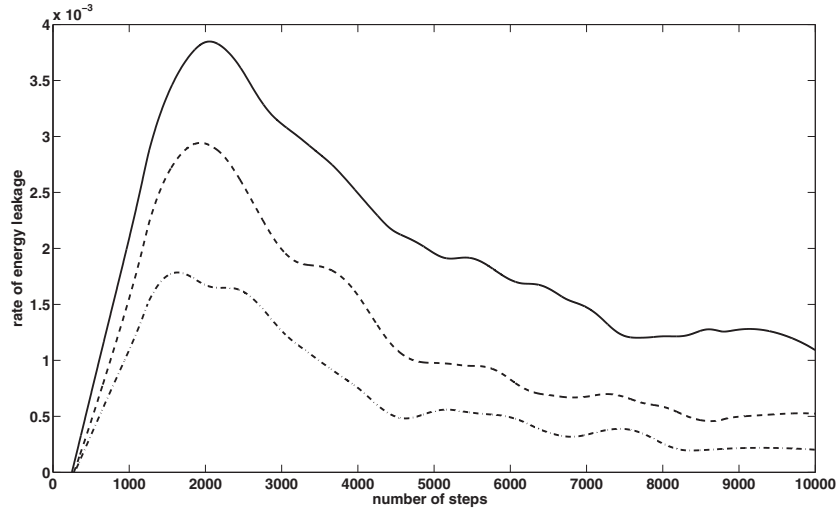


Fig. 2. Rate of energy leakage from the cavity for various values of the mirror reflection probabilities: solid line – $R = 0.25$, dashed line – $R = 0.50$ and dashed-dotted line – $R = 0.75$. We assume the deterministic initial conditions and the probabilities $p_a = 1.00$, $p_e = 0.25$.

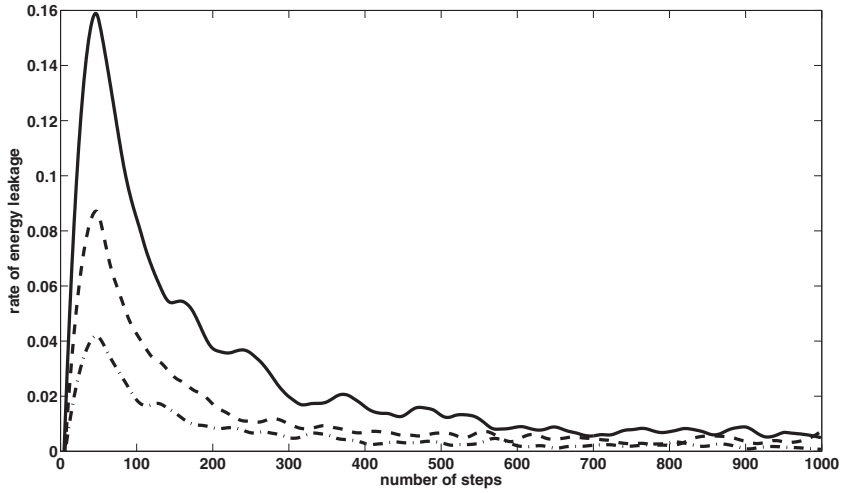


Fig. 3. The same as in Fig.2 but for $p_a = 0.25$, $p_e = 1.00$.

Thus, Fig.2 shows the energy leakage rates for various values of the reflection probability for the mirrors confining the system. We see that the rate r is equal to zero for approx. 300 first steps. This is related to the fact that the excited cells are located in the central part of the automaton. Therefore, the excitations need some period of time to reach the "mirrors". After that the value of r increases rapidly reaching some maximum and begins to fall. This rate decay resembles an asymptotic decay. It means that the system relatively fast loses its energy and after that if there are very few excited cells left, the energy of the whole system does not change a lot. From Fig.2 one can see that the lower the reflection probability R the higher values of the rate r are achieved. Moreover, we see that the length of the initial period of time when $r = 0$ does not depend on the value of R. These features are in a good agreement with the expected behaviour of energy in real cavity systems. This fact strengthens our conviction about the correctness of our model.
Fig.3 shows the situation analogous to that shown in Fig.2 but for $p_a = 0.25$ and $p_e = 1.00$. This means that every cell that is excited for a given time-step emits energy quantum immediately (in the next time-step). Generally, in Fig.3 we see the same features as those shown in Fig.2 but for this case all processes are more rapid in character. As we compare the time-scales of these two plots we see that for the case discussed here we deal with processes one order of magnitude faster then previously. Moreover, the values of the energy leakage rate are two orders of magnitude greater than for the cases shown in Fig.2. This is not only an effect of the immediate energy emission by excited cells. As shown in paper (Kowalewska-Kudłaszyk & Leoński (2008)), absorption of energy by the cells that are in their ground states can play a dominant role in the system. This effect was referred there as to *molasses effect*.

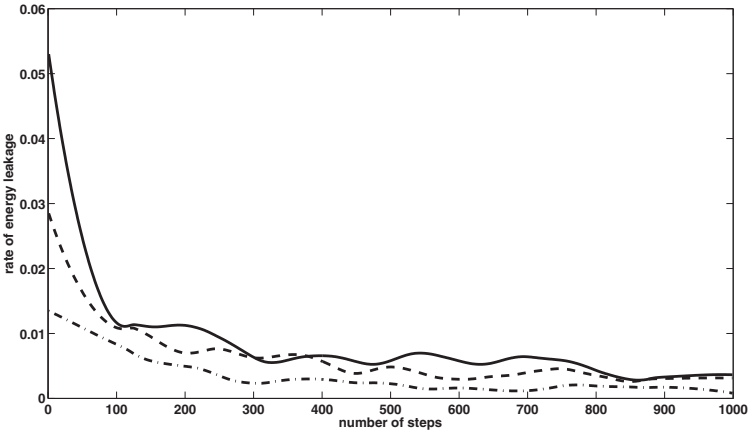


Fig. 4. The same as in Fig.2 but for random initial conditions.

Finally, we show the energy leakage rate r time-dependence for random initial condition. We assume that for the time $t = 0$ we have 50 excited cells again but they are randomly spread across the whole automaton line. All parameters are assumed to be equal to those from Figs.2 and 3. Thus, Fig.4 corresponds to the situation depicted in Fig.2, whereas Fig.5 is related to Fig.3. In both figures (Figs.4 and 5) one can generally see the behaviour of r analogous to that discussed previously (Figs.2 and 3, respectively). However, for the case of random initial

condition, r starts its evolution from non-zero maximal values decreasing with time. We do not observe the initial "dead" period when $r = 0$. Such a behaviour is related to the fact that all excitations are already spread across the automaton line for $t = 0$. As a result, the excitations do not need any initial period of time to reach the cavity border ("mirror").

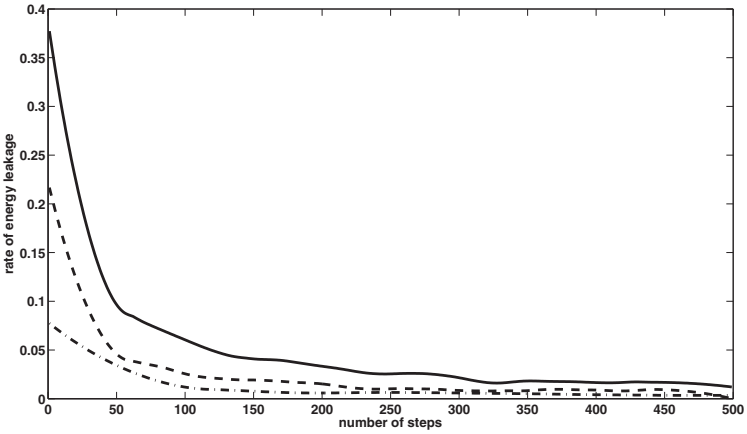


Fig. 5. The same as in Fig.3 but for random initial conditions.

3.2 Entropic disorder measure and autocorrelation function

As we assume that the reflection probability R is equal to 1.00 or(and) we deal with periodic boundary conditions we remove all dissipation effects from our model and the total energy of the system remains constant (the rate $r = 0$). For such a case the model becomes a very good tool for investigation of disorder. If the excitations are initially grouped in a given part of the automaton after some time of evolution they are randomly spread over the whole system. As a result we start from an ordered system and end at disordered one. Obviously, it is possible to include dissipation to the model and we shall do it for the cases discussed here. However, at this point it should be mentioned that if the total number of excitations decreases, some problems with determination of whether the system is ordered or not can occur (this fact was discussed in the paper Kowalewska-Kudłaszyk & Leoński (2008)).

Although it is possible to measure the degree of ordering of our system with application of various parameters, we shall concentrate on that proposed in our earlier papers (Kowalewska-Kudłaszyk & Leoński (2006; 2008)). We have proposed and discussed the entropic parameter $\mathcal{E}$ defined on the basis of the Fourier transform

$$F(\omega) = \int_{x_0}^{x_{end}} f(x)\, \exp(-i\omega x)\, dx \quad , \tag{1}$$

where the function $f(x)$ is equal to zero for the cells in their ground states or to unity for the excited cells. We perform integration over the whole length of the automaton and x labels the cell position. Due to the discrete character of CA we replace this integral by the discrete sum as follows

$$F(\omega) = \sum_{x=x_0}^{x_{end}} f(x)\, \exp(-i\omega x) \quad . \tag{2}$$

Applying thus defined function we introduce the "power spectrum" $P(\omega) = |F(\omega)|^2$. Next, we normalize it and define the "entropy" of the system discussed as

$$\mathcal{E} = \sum_{\omega} P_N(\omega) \log(P(\omega)) \quad , \tag{3}$$

where we perform the summation over the all frequencies appearing in the system. The index "N" appearing here means that the spectrum is already normalized. To avoid interpretation problems related to the influence of the total number of excited cells on the value of entropy we shall deal with the "entropy per number of excited cells". Practically, for each time-step we divide the parameter $\mathcal{E}$ by the current number of excitations in the system.
Thus, Fig.6 shows the time-evolution of $\mathcal{E}$ for various values of the reflection probability R. Moreover, we assume that the absorption probability is small ($p_a = 0.25$) and the excited cells emit energy immediately ($p_e = 1.00$). For the time $t = 0$ all excitations are assumed to be located centrally, *i.e.* we have 50 excited cells at the locations 51-100 (cells 1-50 and 101-150 are in their ground states). From Fig.6 we see that for $R = 0.50$ and 0.75 the time-evolution of R can be divided into two stages. First, we see its rapid growth and then (after approx.100 time-steps) almost linear – very slow increase in the value of R. This slow increase can be practically neglected and the entropy can be treated as constant (for the discussed period of time). However, if we assume that $R = 0.25$ the third, intermediate stage is apparent. For the time-steps $\sim 20 - 300$ we observe the smooth, nonlinear increase in the entropy. After that we can see the linear increase again, however, the increase-rate is considerably greater than for the cases of greater values of R.

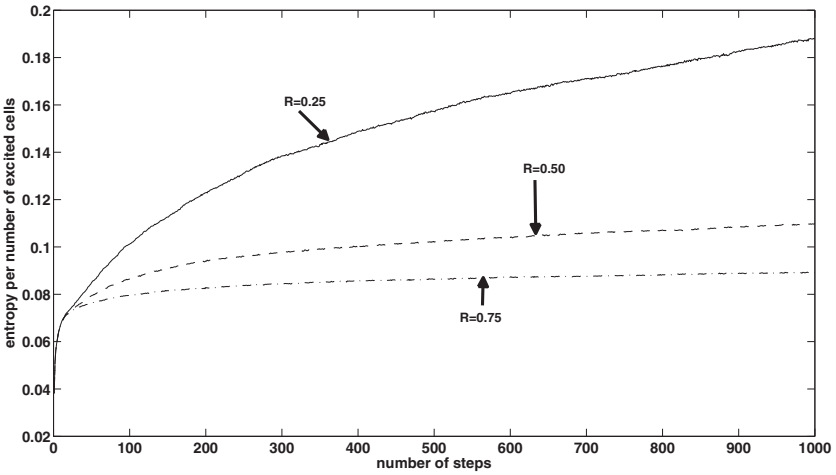


Fig. 6. The entropies for various values of the reflection probability R. Other parameters are: $p_a = 0.25$, $p_e = 1.00$. Random initial conditions are assumed.

Fig.7 shows the opposite, situation – we assume that the absorption probability is equal to unity whereas, $p_e = 0.25$. We see that the first step (rapid, initial growth of the entropy) and the second one (nonlinear growth) are identical for all values of R. However, for the time-steps above ~ 1500 we see a linear increase in entropy but at different rates for different

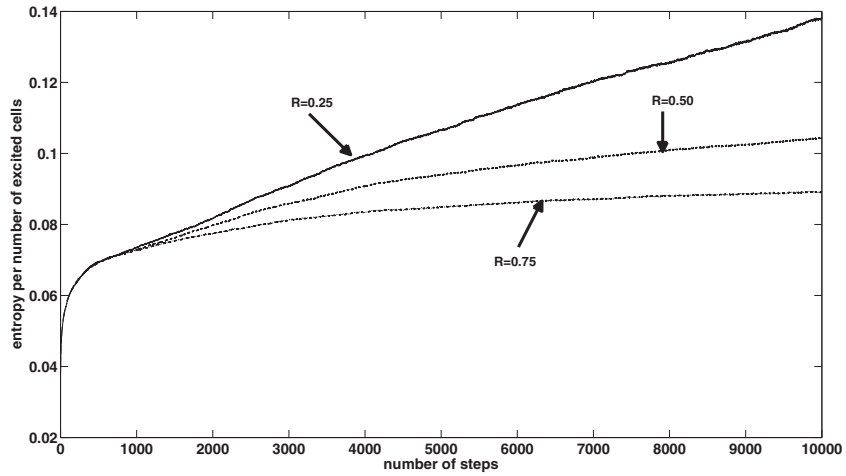


Fig. 7. The same as in Fig.6 but for $p_a = 1.00$ and $p_e = 0.25$.

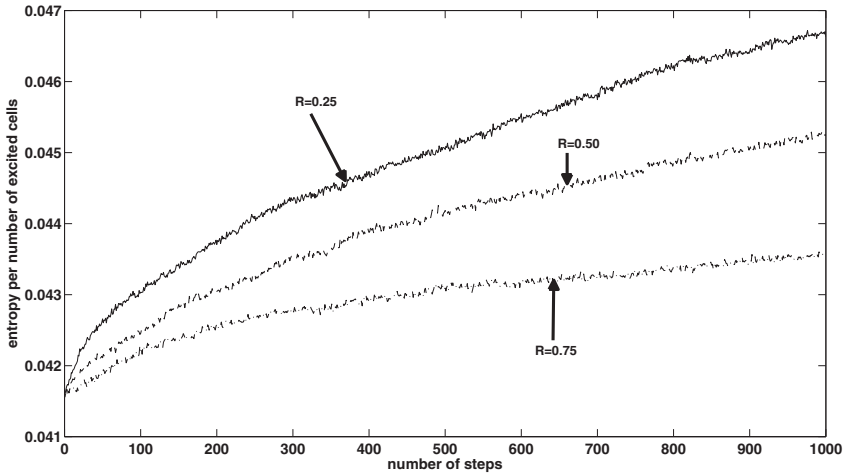


Fig. 8. The same as in Fig.7 but for random initial conditions.

values of the reflection probability. For $R = 0.50$ and 0.75 we observe slowly growing value of $\mathcal{E}$ whereas for $R = 0.25$ the increase is more pronounced. This behaviour is similar to that shown in Fig. 6, but one should note that the time-scale we discuss here is by one order of magnitude greater then previously. This is a result of the fact that the absorption probability is assumed to be unity. For this case the whole system's evolution slows down considerably and this is the mentioned earlier *molasses effect*.

Fig.8 presents the time-evolution of $\mathcal{E}$ for random initial conditions. For this case it is assumed that for $t = 0$, the excitations are randomly distributed along the whole system. As a result, the initial values of entropy are comparable to its final ones, contrary to the previous cases (deterministic initial conditions) where $\mathcal{E}$ started its evolution practically from zero value. Moreover, the entropy time-variations have noisy character and now we shall concentrate on it.

One of the methods for investigation of systems in which some kind of disorder appears is that based on the autocorrelation function (AF). This function, frequently used in signal processing, measures the existence of correlations of the signal with itself but shifted in time scale backwards for a specified time value. The function can confirm the existence of any repetitive patterns in the signal analysed, even if they are obscured by any kind of noise that affects the signal considered.

AF can be used to trace the degree of disorder that emerges during evolution of the system we are dealing with. If our system is initially ordered, we can try to find whether during the evolution under various conditions applied (as absorption and emission probabilities or refraction probability of resonator mirrors), the system becomes disordered or not. If there are such conditions that would force the excitations to spread inside the resonator in any periodic way, the autocorrelation function notes that fact.

The definition of the autocorrelation function used comes from the relation (Bendat & Persol (2010)):

$$C(\tau) = \frac{1}{N-1-\tau} \sum_{i=0}^{N-1-\tau} a_i \cdot a_{i+\tau} \ , \tag{4}$$

where τ is the time delay, N is the number of elements in series and a_i is the value of the $i-th$ element of the series analysed. If the autocorrelation coefficient is to be normalised, it should be additionally divided by the value of C for $\tau = 0$ and in this paper such normalised AF is used.

For the cases discussed here the autocorrelation function will be applied as a measure of correlations between the entropies corresponding to various moments of time. Autocorrelation function measures the character of changes in entropy with time. We shall examine two extreme cases for both random and deterministic initial conditions: the former corresponds to maximal absorption and small emission probabilities (Fig.9 and Fig.11) and the latter to small absorption and maximal emission probabilities (Fig.10 and Fig.12).

We start our consideration from the case when the *molasses effect* is dominant in the system. This situation corresponds to the assumed maximal absorption and relatively small emission probabilities. For deterministic initial conditions (Fig.9) we can see slow decay of AF with increasing time-shift τ parameter, since the entropy is a slowly increasing function of time. It means that the system during its evolution becomes more disordered than it was at the beginning, but this process is rather slow. In contradiction to the opposite probability values (Fig.10 – $p_a = 0.25$ and $p_e = 1.00$) we can observe practically no changes in longer time. The reason is that the entropy reaches its high value almost instantly and the system initially ordered very quickly becomes disordered. After that any changes in $\mathcal{E}$ are rather slow in character as we compare them with its initial growth. For this case we can differentiate AF corresponding to various values of R much more easily than for the previous case. This is a result of the fact that for this case AF is almost constant and when the value of τ becomes

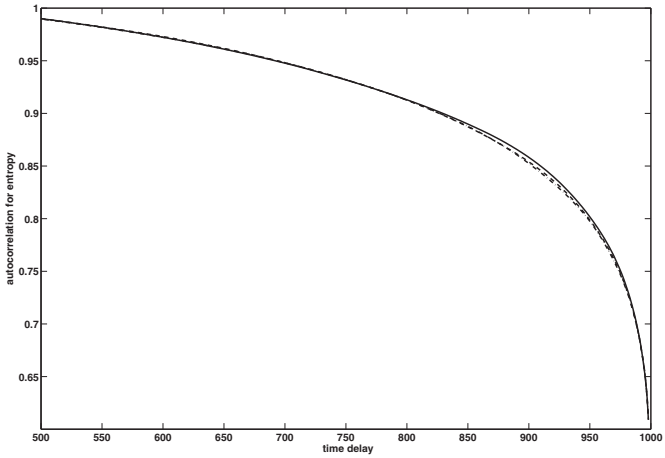


Fig. 9. Entropy autocorrelation function v. time-delay parameter τ, for various values of the reflection probability R: solid line – $R = 0.25$, dashed line – $R = 0.50$ and dashed-dotted line –$R = 0.75$. Other parameters are: $p_a = 1.00$, $p_e = 0.25$. Deterministic initial conditions are assumed.

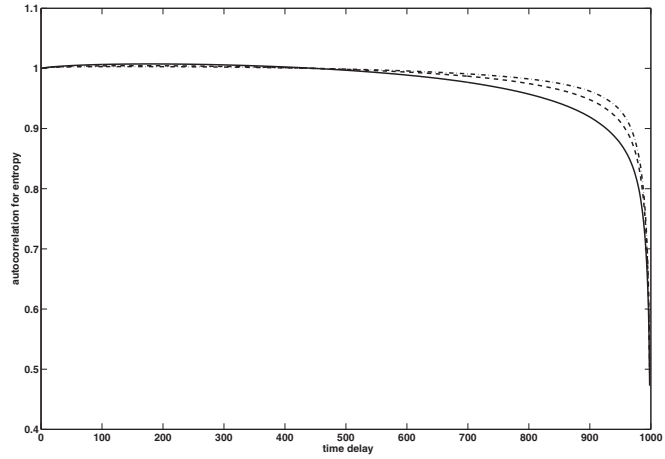


Fig. 10. The same as in Fig.9 but for $p_a = 0.25$, $p_e = 1.00$ and various values of R: solid line – $R = 0.25$, dashed line – $R = 0.50$, dashed-dotted line – $R = 0.75$.

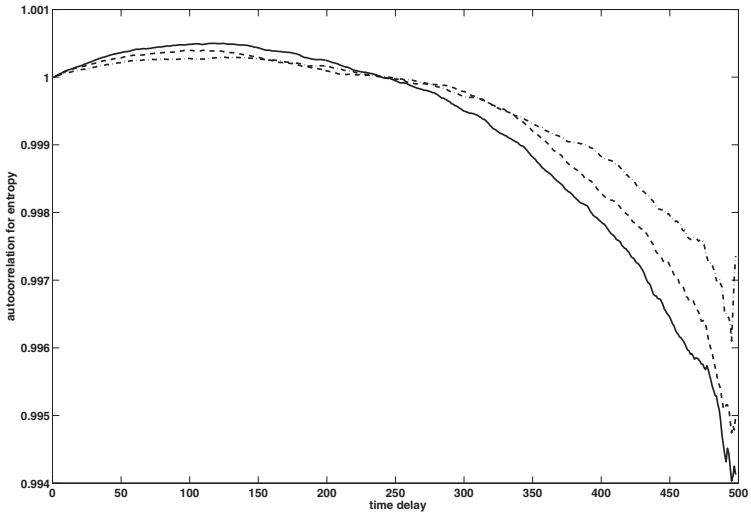


Fig. 11. The same as in Fig.9 but for random initial conditions.

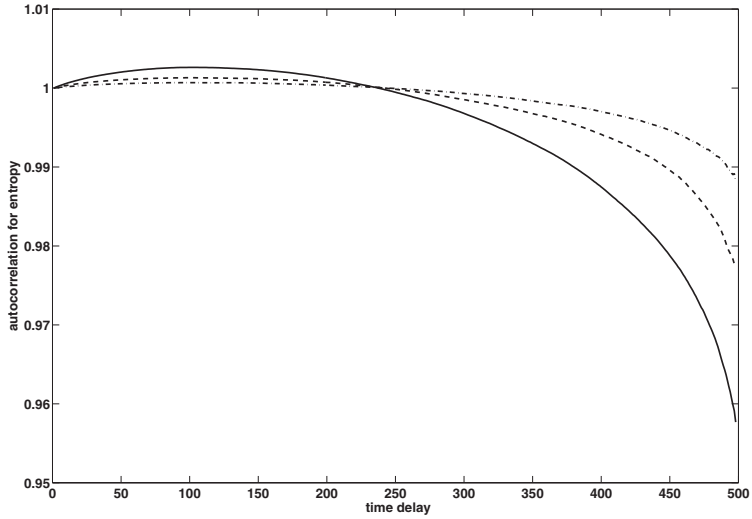


Fig. 12. Entropy autocorrelation function for the same parameters as those of Fig.10 but with random initial conditions.

close to the cavity dimension this function falls rapidly. This decrease appears more suddenly and is more rapid as the probability of reflection increases. Such behaviour is caused by greater velocity of the excitation movement – the excitations reach the "mirrors" faster than it was for maximal absorption probability case. This leads to the visible differences in the dynamics of energy leakage from the cavity.
The analogous features can be seen for the random initial conditions cases (Fig.11 and 12), but the changes in AF are very tiny as we compare them with those corresponding to the ordered, deterministic initial conditions. One should remember that the excited cells are initially spread along the whole cavity. Even when the absorption probability reaches its maximal value, the energy quanta can escape from the cavity sooner than it was for the case of deterministic initial conditions. For both values of absorption and emission probabilities we can also see some initial small growth in AF but one should keep in mind that the vertical axes scale appearing in Figs.11 and 12 is much more stretched as we compare it with those in Fig.9 and 10. However, such small increase in the AF value is an effect of short-time entropy vs. time regular dependence at the beginning of the entropy growth process. However, after this short period of time, entropy becomes an irregular function of time and AF decreases.

3.3 Recurrence plots

As shown, the system discussed here can exhibit complicated evolution. To determine its character, we can apply the *recurrence plots* (RP) analysis. This investigation method allows determination whether the system evolves chaotically, exhibits regular or periodic dynamics or finally, is subjected by some noise. Moreover, RP can be applied for the situations when we cannot obtain sufficiently long time-series. For explanation of the RP idea one can see the review paper Marwan et al. (2007) *and the references quoted therein*. The idea of RP was proposed by Eckmann et al. (1987) and is widely used in nonlinear data analysis up to now.
To build RP we need to find the binary matrix. Its elements are defined by the formula (Eckmann et al. (1987)):

$$R_{i,j} = \Theta(\epsilon_{thr} - ||\vec{x}_i - \vec{x}_j||), \qquad i,j = 1,\ldots,N, \qquad \vec{x}_i, \vec{x}_j \in \mathcal{R} \ , \tag{5}$$

where Θ is the Heaviside function, ϵ_{thr} is the threshold parameter, and $||\ ||$ denotes the norm. This norm allows determination of the distance between two points. To determine this matrix we need to reconstruct the trajectory in the phase-space on the basis of the time-series we investigate. The modulus $||\vec{x}_i - \vec{x}_j||$ determines the distance between the points in the reconstructed phase-space. If two points fall inside the same region (sphere of the radius ϵ_{thr}) they are labelled by 1, otherwise they labelled by 0. Applying this procedure we obtain a matrix filled with zeros and ones that can be illustrated by white and black point. If we reconstruct the phase space in which the system analysed lives, we look for times at which system returns to the same area of that phase space. In fact, RP shows whether the system analysed via the time series inspection recurs or not. Analysis of the so obtained matrix (picture consisting of the black and white points) allows determination the dynamics we are dealing with.
In this chapter we present RP for automaton determined by random initial conditions where the excited cells are randomly spread over the whole system. Moreover, we shall discuss the cases of the cavity with cyclic boundary conditions and that of the cavity confined by the "mirrors". Since, the energy of such a cyclic model is preserved we shall concentrate at this point on the entropic parameter $\mathcal{E}$ time-evolution. This "entropy" describes how disordered our system is and RP based entropy analysis gives us information about the character of

changes in $\mathcal{E}$. The type of such variations should reflect the character of the whole system's dynamics. Excitation movements along the "cavity" are determined by the absorption and emission probabilities that influence the derived RP as well. RP derived on the basis of our model should clarify the nature of the system evolution despite the complex form of this evolution. Many features are obscured by the random processes taking place in the system and we shall show that RP can reveal them.

At first, let's concentrate on the cyclic boundary conditions and compare two extreme cases: maximal absorption with small emission and maximal emission with small absorption probabilities. The time-evolutions of entropy and the corresponding RP for all cases discussed in this section are plotted. The character of evolution of $\mathcal{E}$ is similar to those discussed in the previous section (Figs.15 and 16) if we assume that the system is confined by the mirrors. For cyclic boundary conditions, the entropy exhibits random deviations from some constant value. If we neglect them the entropy can be treated as constant (note the scale of vertical axis in Figs.13 and 14). However, the question arises whether these random deviations originate from noise effects or some deterministic chaotic features can be expected. Moreover, it would be desirable to check whether any quasi-periodic or periodic evolution are hidden behind the "noisy" view of $\mathcal{E}$ evolution. Therefore, these entropies are plotted with the corresponding RPs.

As follows from the recurrence plot shown in Fig.13 ($p_a = 1.00$, $p_e = 0.05$), the entropy evolution has a rather noisy than chaotic character. Although some diagonal lines appear, but they are formed of a few points only (average: 2.5 points). Since the length of the diagonal lines is related to the value of the sum of positive Lyapunov exponents, short lines indicate that we have not chaotic behaviour in this case. We see that in the system the noise effects are dominant over the chaotic ones. This observation agrees with the fact that for the situation discussed here, the fraction of all of the points that form the diagonal lines is about 8%. This indicates that a significant number of the points that recur are practically isolated ones. All these features confirm that the entropy changes are of noisy character. For the case when $p_a = 0.05$, $p_e = 1.00$ (see Fig.14) again the entropy vs. time-evolution of $\mathcal{E}$ should be classified as a noisy signal. For this case the percent of points forming diagonal lines is smaller than it was observed previously. It means that for both extreme cases initial random conditions result in a noisy character of entropy dynamics.

Next, we change the boundary conditions from the cyclic ones to those corresponding to the cavity with mirrors. Although for this case the dissipation processes is included, we shall discuss the character of the entropy evolution again. This allows a comparison of the results with those discussed above for the cyclic conditions. Fig.15 shows the entropy evolution and the corresponding RP for the probability of reflection $R = 0.75$. Moreover, it is assumed that $p_a = 1.00$, $p_e = 0.10$ and the cells are initially randomly excited. The plot of $\mathcal{E}(t)$ (Fig.15 – top) has the same irregular character as that for the cyclic conditions. The only difference is in the initial growth of entropy. Nevertheless, from the form of this plot we are not able to say anything on the character of the time-evolution. However, analysis of RP (Fig.15 – bottom) shows that the dynamics of the system differs considerably from that discussed earlier. Both horizontal and diagonal lines forming rectangular structures are observed. They are dominant over single dots characteristic of noise, and they are a result of some periodic and quasi-periodic effects and drift ones (logistic map is corrupted with a linearly increasing term). Moreover, the plot shows some features characteristic of the Brownian motion. Periodic effects are related to the finite length of the cavity and a finite constant velocity of the excitation

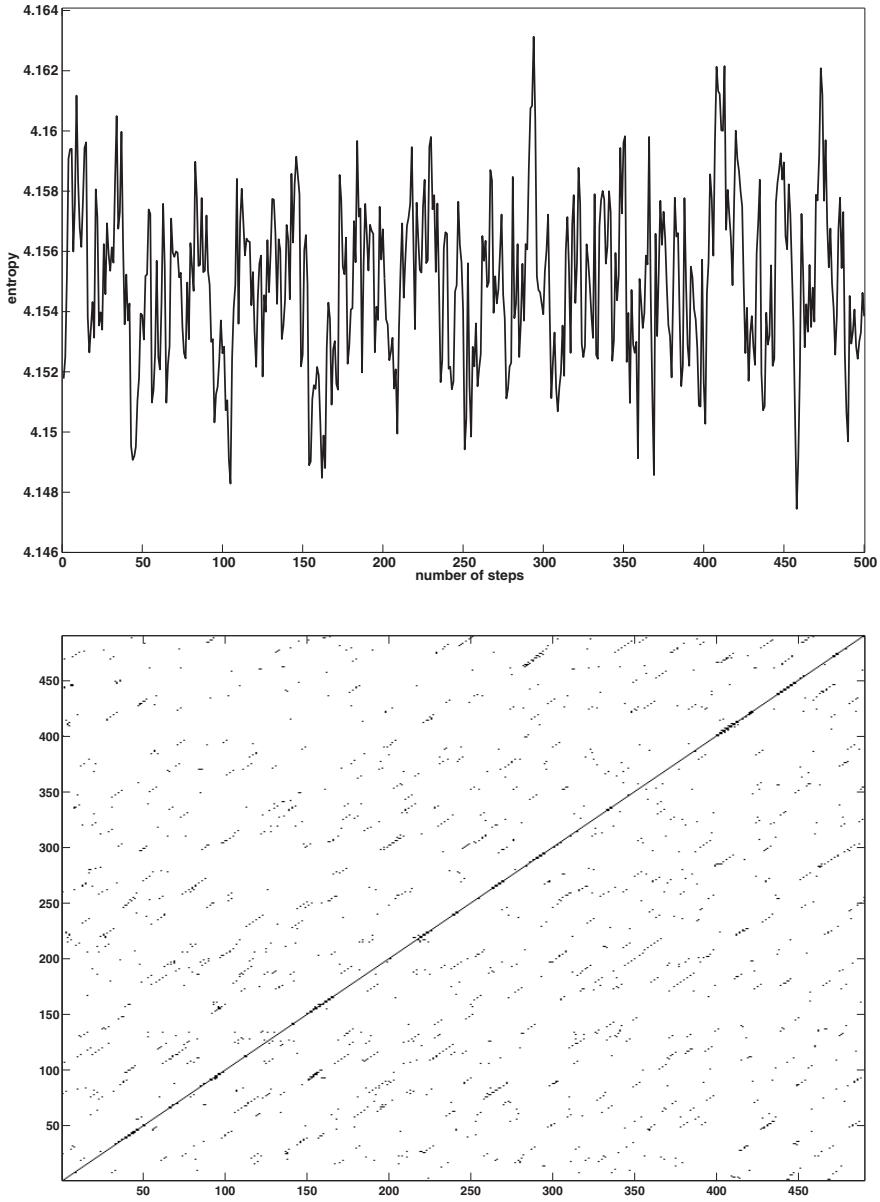


Fig. 13. At the top the entropy for $p_a = 1.00$ and $p_e = 0.05$ and cyclic boundary conditions. At the bottom the recurrence plot corresponding to the entropy evolution. We assume random initial conditions.

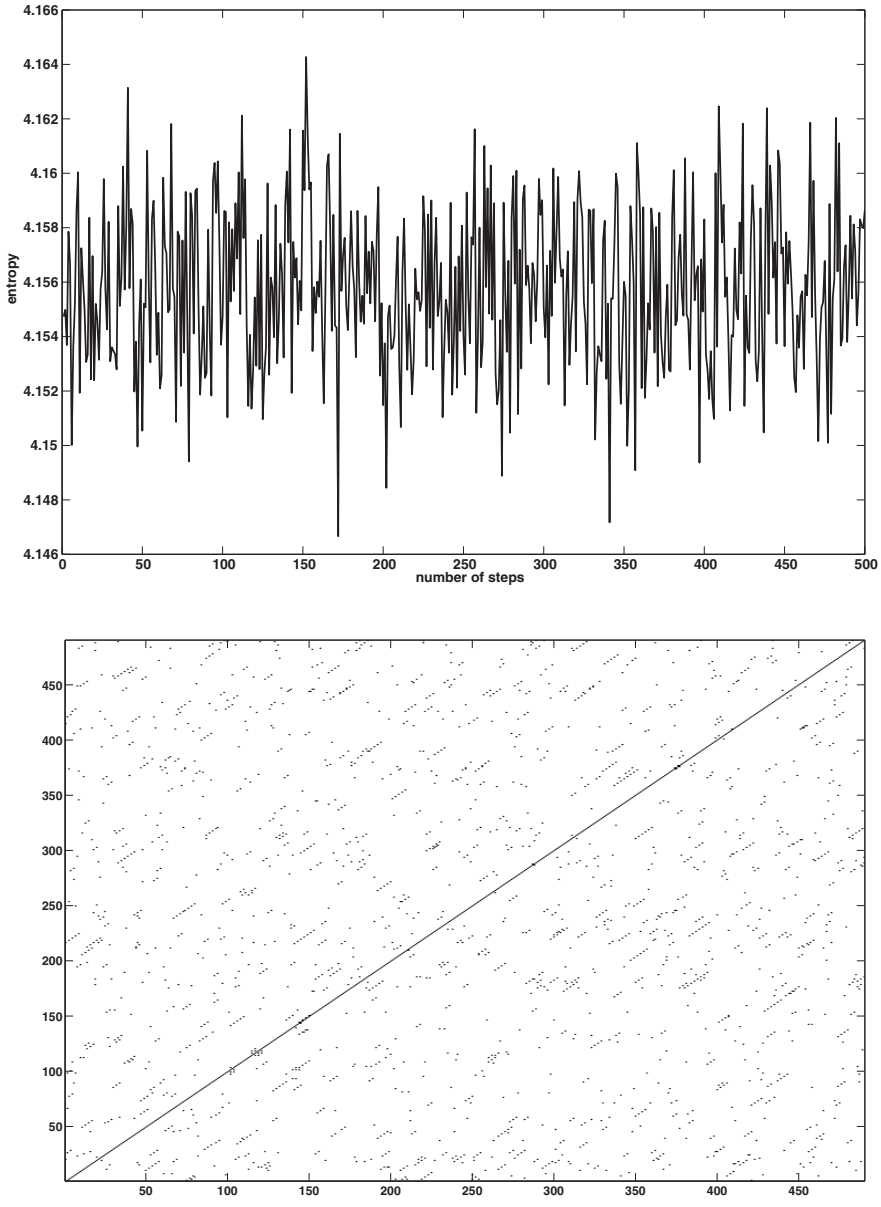


Fig. 14. The same as in Fig.13 but for $p_a = 0.05$ and $p_e = 1.00$.

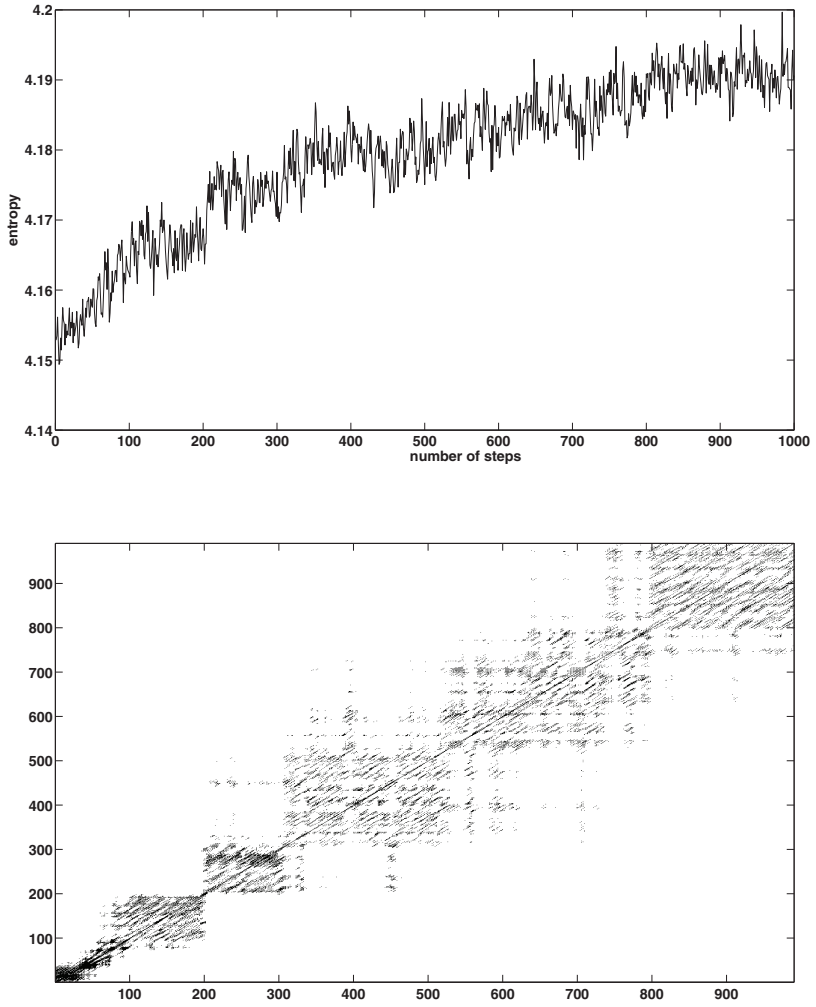


Fig. 15. The same as in Fig.13 but for the cavity with mirrors, $R = 0.75$, $p_a = 1.00$ and $p_e = 0.10$.

movement (the excitations "oscillate" from one end of the cavity to another). The leakage of the energy from the system moderates the noise effects and permit uncovering of other features. Moreover, for this case the time-evolution becomes more regular than for the case of constant energy, when many excitations moved randomly inside the cavity.
If energy leakage is increased ($R = 0.25$), the time-evolution of $\mathcal{E}$ looks similar to that discussed for $R = 0.75$ (Fig.16 – bottom). However, RP form indicates different character of this evolution. We see that horizontal lines forming rectangular structures again, however

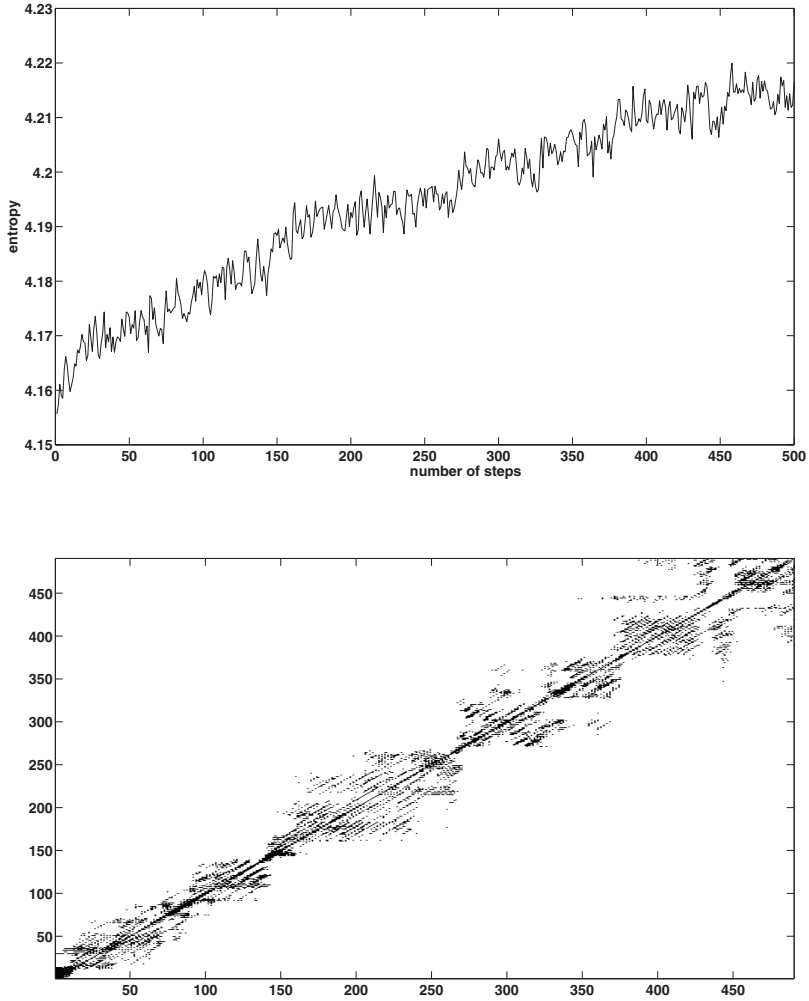


Fig. 16. The same as in Fig.15 but for $R = 0.25$.

they are much closer to the main diagonal then in the former case. This means that the system has stabilized itself and there are some regular oscillations inside the cavity. Moreover, there are some diagonal lines, but they are very short and sparse. In fact, during the evolution many energy quanta escapes the system, so fewer excitations remain inside the cavity and their oscillations becomes more distinct.

The results have shown that CA evolution described by RP can exhibit many interesting features. Thanks to the application of RP, we can detect and classify various types of the

system-evolutions that are invisible and usually remain obscured by a seemingly chaotic behaviour of the system.

4. Final remarks

We have shown how the simple rules determining the CA evolution character can lead to a complicated evolution of the model considered. The model discussed here exhibits various features characteristic of the real cavity system. We proposed application of various parameters and investigation methods to characterize the dynamics of the model and hence, its physical properties.

Our model can be treated as a starting point for further investigation. For instance, two or more dimensional systems can be considered instead of one-dimensional one discussed here. Moreover, other mechanisms of energy dissipation than by the "mirrors" considered in this work, and excitation processes also can be taken into regard. However, it should be emphasized that despite the simplicity of the model proposed, it turned out to be a fruitful tool for dynamics investigation.

5. References

Allen L. & Eberly J. H. (2007) *Optical Resonance And Two-Level Atoms*, Dover Publications, Inc., ISBN 9780486655338, Mineola, New York

Barclay M., Andersen H. & Simon C. (2010). Emergent Behaviors in a Deterministic Model of the Human Uterus, *Reproductive Sciences*, Vol. 17, No. 10, 948-954

Bendat J. S. & Persol A. G. *Random Data - Analysis and Measurement Procedures*, J. Wiley & Sons, Inc., ISBN 978-0-470-24877-5, Hoboken, New Jersey

Butler, J. T (1973). A note on cellular automata simulations, *Information and Control*, Vol. 26, No. 3, 286-295

Cleveland W. S. (1979). Robust locally-weighted re- gression and smoothing scatterplots, *J. Am. Stat. Assoc.*, Vol. 74, No. 368, 829-836

Clevlelend W. S. & Devlin S. J. (1998) Locally Weighted Regression: An Approach to Regression Analysis by Local Fitting *Journal of the American Statistical Association* Vol. 83 No. 403, 596-610

Combinido J. S. L. & Lim M. T. (2010). Modeling U-turn traffic flow, *Physica A*, Vol. 389, No. 17, 3640-3647

Dabaghian V., Jackson P., Spicer V & Wuschke K. (2010). A cellular automata model on residential migration in response to neighborhood social dynamics, *Math. Comp. Modell.* Vol. 52, No. 9-10, 1752-1762

Eckmann J.-P., Kamphorst S. O. & Ruelle D. (1987). Recurrence Plots of Dynamical Systems, *Europhys. Lett.* Vol. 4, No. 9, 973-977

Guisado J. L., Jimenez-Moralez F. & Guerra J. M. (2003). Cellular automaton model for the simulation of laser dynamics, *Phys. Rev. E*, Vol. 67, No. 6, 066708

Kondrat G. & Sznajd-Weron K. (2010). Spontaneous Reorientations in a Model of Opinion Dynamics with Anticonformists, *Int. J. Mod. Phys. C*, Vol. 21, No. 4, 559-566

Kowalewska-Kudłaszyk A. & Leoński W. (2006). Two-Level Systems, their Evolution and Cellular Automata Method, *Acta Phys. Hung. B*, Vol. 26, No. 3-4, 247-252

Kowalewska-Kudłaszyk A. & Leoński W. (2008). Cellular automata and two-level systems dynamics – Spreading of Disorder, *J. Comp. Meth. Sci. Eng.*, Vol.8, No. 1-2, 147-157.

Ladd A. C. & Colvin M. E. (1988). Application of lattice-gas cellular automata to the Brownian motion of solids in suspension, *Phys. Rev. Lett.*, Vol. 60, No. 11, 975-978

Lejeune A., Perdang J. & Richert J. (1999). Application of cellular automata to N-body systems, *Phys. Rev. E*, Vol. 60 No. 3, 2601-2611

Margolus N., Toffoli T. & Vichniac G. (1986). Cellular-Automata Supercomputers for Fluid-Dynamics Modeling, *Phys. Rev. Lett.*, Vol. 56, No. 16, 1694-1696

Marwan N., Romano M. C., Thiel M. & Kurths J. (2007). Recurrence Plots for the Analysis of Complex Systems, *Physics Reports*, Vol.438, No. 5-6, 237-329.

Kondrat G. & Sznajd-Weron K. (2009) Spontaneous Reorientations in a Model of Opinion Dynamics with Anticonformists, *Int. J. Mod. Phys. C*, Vol.21 No. 4, 559-566

Rosin P. L. (2010). *Comp. Vision Image Understanding*, Vol. 114, No. 7, 790-802

Vichniac G., (1984). Simulating Physics with Cellular Automata. *Physica D*, Vol.10, No. 1-2, 96-116

Walczak M. & Leoński W. (2003) A cavity with two-level atoms and cellular automata, *Fortchr. Phys.*, Vol. 51, No. 2-3, 186-189

Zeng H. C., Pukkala T., Peltola H. & Kellomaki S. (2010) Optimization of irregular-grid cellular automata and application in risk management of wind damage in forest planning, *Can. J. Forrest Res.*, Vol. 40, No. 6, 1064-1075

21

Cellular Automata Simulation of Two-Layer Ising and Potts Models

Mehrdad Ghaemi
Tarbiat MoallemUniversity
Iran

1. Introduction

One of the most interesting phenomena in the physics of the solid state is ferromagnetism. Ferromagnetism and antiferromagnetism are based on variations of the exchange interaction, which is a consequence of the Pauli principle and the Coulomb interaction. In the simplest case of the exchange interaction of two electrons, two atoms or two molecules with the spins σ_1 and σ_2, the interaction has the form $E = -J\sigma_1\sigma_2$, where J is a coupling constant which depends on the distance between the spins. When the coupling constant is positive, then a parallel spin orientation is favored. This leads in a solid to ferromagnetism. This happens, however, only when the temperature is lower than a characteristic temperature known as the Curie temperature. Above the Curie temperature the spins are oriented at random, producing no net magnetic field (Fig. 1). As the Curie temperature is approached from both sides the specific heat of the metal approaches infinity.

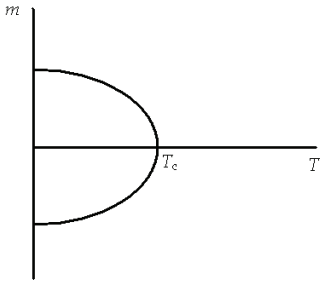


Fig. 1. Temperature dependence of magnetization

When the coupling constant is negative, then an antiparallel spin orientation is preferred. In a suitable lattice structure, this can lead to an antiferromagnetic state. The exchange interaction is short-ranged; but owing to its electrostatic origin it is in general considerably stronger than the dipole-dipole interaction. Examples of ferromagnetic materials are Fe, Ni, EuO; and typical antiferromagnetic materials are MnF2 and RbMnF3.
The Ising model is a crude attempt to simulate the structure of a physical ferromagnetic substance. This model plays a very special role in statistical mechanics and generates the

simplest nontrivial example of a system undergoing phase transitions. Its analysis has provided us with deep insights into the general nature of phase transitions which are certainly better understood nowadays after the publication of the hundreds of papers which followed the pioneering work of Onsager (Onsager, 1944).

Although, at zero magnetic field, there is an exact solution for the 2-dimensional (2-D) Ising model (Onsager, 1944 and Huang, 1984), however, there is no such a solution for the two-layer Ising and Potts models. The Potts models are the general extension of the Ising model with q-state spin lattice i.e., the Potts model with $q = 2$ is equivalent to the Ising model. Although we do not know the exact solution of the two-dimensional Potts model at present time, a large amount of the numerical information has been accumulated for the critical properties of the various Potts models. For further information, see the excellent review written by Wu (Wu, 1982) or the references given by him.

The two-layer Ising model, as a simple generalization of the 2-D Ising model has of long been studied (Ballentine, 1964; Allan, 1970; Binder, 1974; and Oitmaa & Enting, 1975). The two-layer Ising model as a simple model for the magnetic ultra-thin film has various possible applications to real physical materials. For example, it has been found that capping PtCo in TbFeCo to form a two-layer structure has applicable features, for instance, raising the Curie temperature and reducing the switching fields for magneto optical disks (Shimazaki et al., 1992). The Cobalt films grown on a Cu (100) crystal have highly anisotropic magnetization (Oepen et al., 1990) and could be viewed as layered Ising models. From the theoretical viewpoint, the two-layer Ising model as an intermediate between 2-D and 3-D Ising models, is important for the investigation of crossover from the 2-D Ising model to the 3-D Ising model. In particular, it has been argued that the critical point of the latter could be found from the spectrum of the 2-layer Ising model (Wosiek, 1994). In recent years, some approximation methods have been applied to this model (Angelini et al., 1995; Horiguchi et al., 1996; Angelini et al., 1997 and Lipowski & Suzuki, 1998). It is also argued that the two-layer Ising model is in the same universality class as the two dimensional Ising model (Li et al., 2001).

Since the exact solution of the Ising model exists only for the one- and two-dimensional models, the simulation and numerical methods may be used to obtain the critical data for other models. One of the numerical methods is using the transfer matrix and decreasing the matrix size (Ghaemi et al., 2004). Ghaemi et al. have used the transfer matrix method to construct the critical curve for a symmetric two-layer Ising model. In another work (Ghaemi et al., 2003), they have used this method to get the critical temperature for the anisotropic two-layer Ising model. Such calculations are limited to lattice with the width 5 cells in each layer and the critical point is obtained by the extrapolation approach. There are other numerical methods for solving the Ising models.

However, the numerical methods mentioned above are time consuming and advanced mathematics is required when they may be used for extended models like the anisotropic two-layer Potts model. In most cases, simulation methods are simple and fast. They are also less restricted to the lattice sizes. There are different simulation methods which have been used to describe Ising and Potts models. Monte Carlo is one of the simulation methods which has been widely used for studying Ising models (Zheng, 1998). In addition, the multicanonical Monte Carlo studies on Ising and Potts models are highly used in recent years (Janke, 1998 and Hilfer et al., 2003). The Cellular Automata (CA) are one of methods that could be used to describe the Ising model. The CA are discrete dynamic systems with simple evolution rules that have been proposed as an efficient alternative for the simulation

of some physical systems. There are some different approaches which are based on the CA method. The Q2R automaton is an approach which is used for the microcanonical Ising model. It is deterministic, reversible and nonergodic and also very fast method. Many works have been performed based on this model (Stauffer, 2000; Kremer & Wolf, 1992; Moukarzel & Parga, 1989; Stauffer, 1997; Glotzer & Stauffer, 1990; Zabolitzky & Herrmann, 1988; and MacIsaac, 1990). Although the Q2R model is deterministic and hence is fast, it was demonstrated that the probabilistic model of the CA like Metropolis algorithm (Metropolis et al., 1953) is more realistic for description of the Ising model even though the random number generation makes it slower. There is a main different between the Cellular Automata (CA) method and the Monte-Carlo method which is in the updating of a system in each step. In the Monte-Carlo method, only one site or a cluster which is randomly chosen is updated in each step. However, in the CA, all sites are updated in each time step without a random selection.In addition to the Monte Carlo method, it was proposed that the Cellular Automata (CA) could be a good candidate to simulate the Ising models (Domany & Kinzel , 1984).

In the last two decade a large amount of works were done for describing Ising models by the CA approach and a great number of papers and excellent reviews were published (MacIsaac, 1990; Creutz, 1986; Toffoli & Margolus, 1990; Kinzel, 1985; and Aktekin, 1999). Most of the works that have been done until now are focused on the qualitative description of various Ising and Potts models or to introduce a faster algorithm. For example, the Q2R automaton as a fast algorithm was suggested which has been studied extensively (Vichniac, 1984; Pomeau, 1984; Herrmann, 1986; Glotzer et al., 1990; Moukarzel & Parga, 1989; and Jan, 1990). It was so fast, because no random numbers must be generated at each step. But in the probabilistic CA, like Metropolis algorithm, generation of the random number causes to reduce the speed of calculation, even though it is more realistic for describing the Ising model.

2. Isotropic two-layer Ising model

Consider a two-layer square lattice with the periodic boundary condition, each layer with r rows and p columns. Each layer has then $r \times p$ sites and the number of the sites in the lattice is $2 \times r \times p = N$. We consider the next nearest neighbor interactions as well, so the number of neighbor for each site is 5. In the two-layer Ising model, for any site we define a spin variable $\sigma^{1(2)}(i,j) = \pm 1$ in such a way that $i = 1,...,r$ and $j = 1,...,p$ where superscript 1(2) denotes the layer number. We include the periodic boundary condition as

$$\sigma^{1(2)}(i+r,j) = \sigma^{1(2)}(i,j) \tag{1}$$

$$\sigma^{1(2)}(i,j+p) = \sigma^{1(2)}(i,j) \tag{2}$$

The configuration energy for this model may be defined (Ghaemi et al., 2003) as:

$$\frac{E(\sigma)}{kT} = -\sum_{i=1}^{r,*}\sum_{j=1}^{p,*}\sum_{n=1}^{2}\{K_x\sigma^n(i,j)\sigma^n(i+1,j) + K_y\sigma^n(i,j)\sigma^n(i,j+1)\} - K_z\sum_{i=1}^{r}\sum_{j=1}^{p}\sigma^1(i,j)\sigma^2(i,j) \tag{3}$$

where * indicates the periodic boundary conditions (eqs 1,2), and Kx and Ky are the nearest-neighbor interactions within each layer in the x and y directions, respectively, and Kz is the interlayer coupling.
Therefore, the configuration energy per spin is

$$e = \frac{E(\sigma)}{kTN} \tag{4}$$

The average magnetization of the lattice for this model can be defined (Newman & Barkema, 2001) as

$$\langle M \rangle = \left\langle \sum_{i=1}^{r,*} \sum_{j=1}^{p,*} \sum_{n=1}^{2} \sigma^n(i,j) \right\rangle \tag{5}$$

and the average magnetization per spin is

$$\langle m \rangle = \frac{\langle M \rangle}{N} \tag{6}$$

The magnetic susceptibility per spin (χ) and specific heat per spin (C) is defined as

$$\frac{\partial < M >}{\partial \beta} = \beta(< M^2 > - < M >^2) \tag{7}$$

$$\chi = \frac{\beta}{N}\left(\langle M^2 \rangle - \langle M \rangle^2\right) = \beta N\left(\langle m^2 \rangle - \langle m \rangle^2\right) \tag{8}$$

$$C = \frac{k\beta^2}{N}\left(\langle E^2 \rangle - \langle E \rangle^2\right) = k\beta^2 N\left(\langle e^2 \rangle - \langle e \rangle^2\right) \tag{9}$$

where $\beta = \frac{1}{kT}$.

In the present work, we considered the isotropic ferromagnetic and symmetric case i.e. $K_x=K_y=K_z=K \geq 0$. We have used a two-layer square lattice with 2500×2500 sites in each layer with the periodic boundary condition. The Glauber method (Glauber, 1963) was used with checkerboard approach to update sites. For this purpose the surfaces of two layers are checkered same as each others. For updating the lattice, we use following procedure: after updating the first layer, the second layer could be updated.
The updating of the spins is based on the probabilistic rules. The probability that the spin of one site will be up (p_i^+) is calculated from

$$p_i^+ = \frac{e^{-\beta E_i^+}}{e^{-\beta E_i^+} + e^{-\beta E_i^-}} \tag{10}$$

where

$$\begin{aligned} E_i^{\pm} = -K\{&\sigma^n(i,j)\sigma^n(i+1,j) + \sigma^n(i,j)\sigma^n(i-1,j) + \sigma^n(i,j)\sigma^n(i,j+1) \\ &+ \sigma^n(i,j)\sigma^n(i,j-1) + \sigma^n(i,j)\sigma^{n'}(i,j)\} \end{aligned} \tag{11}$$

and

$$\sigma^n(i,j) = +1 \text{ for } E_i^+$$
$$\sigma^n(i,j) = -1 \text{ for } E_i^- \tag{12}$$

and $\sigma^{n'}(i,j)$ is the neighboring site (*i,j*) in the other layer. Hence, the probability that the spin to be down is

$$p_i^- = 1 - p_i^+ \tag{13}$$

The approach is as follow: first a random number is generated. If it is less than p_i^+, the spin of the site (*i,j*) is up, otherwise (it means that random number is greater than p_i^+), it will be down.

When we start CA with the homogeneous initial state (namely, all sites have spin up or +1), before the critical point (K_C), the magnetization per spin (m) will decay rapidly to zero and fluctuate around it. After the critical point, m will approach to a nonzero point and fluctuate around it; and with increasing of K, the magnetization per spin will increase. But at the critical point, m will decay very slowly to the zero point and the fluctuation of the system will reach to a maximum. For each K, the time that m reaches to the special point and starts to fluctuate around it is called the relaxation time (τ). On the other words, the relaxation time is the time that the system is thermalized. The value of τ can be obtained from the graph of m vs. t (Fig. 2).

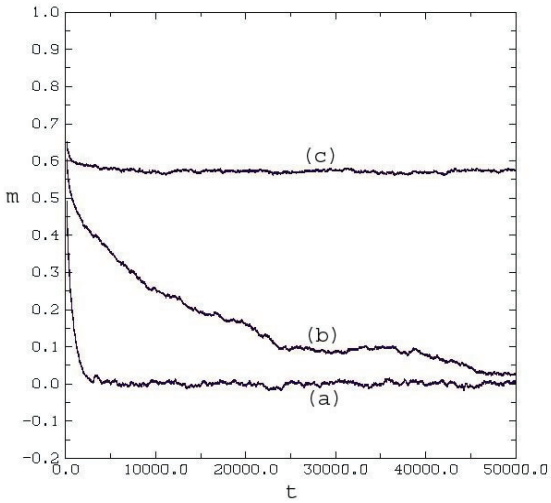


Fig. 2. The magnetization versus time in the two-layer Ising model. for 3 states. a: K=0.304 ($K<K_c$), τ =3500. b: K=0.310 ($K=K_C$), τ =46000. c: K=0.313 ($K>K_c$), τ =4000. (each layer has 2500×2500 sites, start from homogeneous initial state "all +1", time steps = 50000)

One can see from these graphs that the relaxation time increases before critical point and reaches to a maximum at K_C, but after the critical point, τ decreases rapidly. So, in the

critical point, the system last a long time to stabilized. Hence, the critical point may be obtained from the graph of τ vs. K (Fig. 3). The obtained critical point from this graph is 0.310 for the two-layer Ising model.

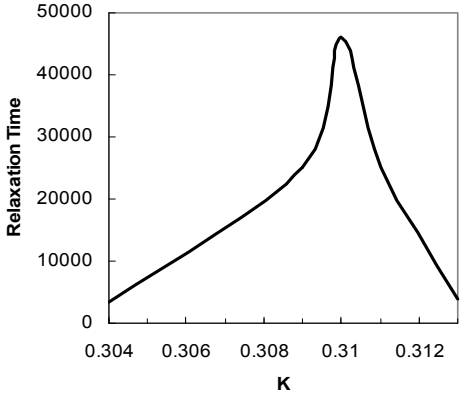


Fig. 3. The relaxation time obtained from Figure 1 versus K for the two-layer Ising model. The maximum appears at K=Kc

In our approach, we have calculated the thermodynamic quantities after thermalization of the lattice. In other words, first we let the system reaches to a stable state after some time step ($t=\tau$), and then to be updated up to the end of the automata (t=50000). For example to calculate the average value of magnetization per spin (<m>), one should add all values of m from the relaxation time up to the end of the automata (or end of the time step) and divide the result to number of steps. The other way for calculation of the critical point is the usage of <m>. By drawing the graph of <m> vs. K, we may also obtain K_C. Fig. 4 shows the results

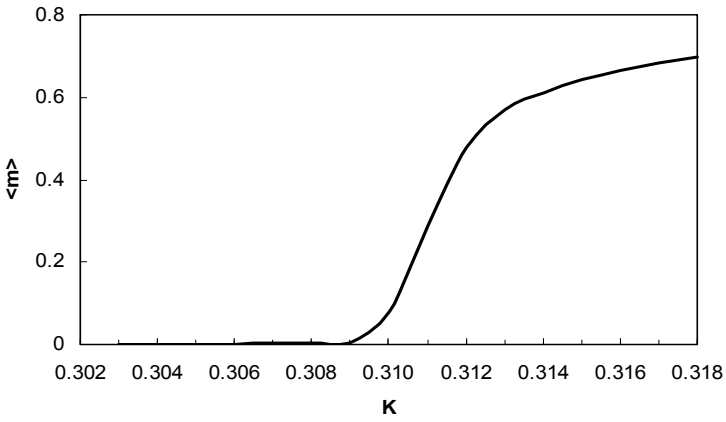


Fig. 4. <m> versus coupling coefficient (K) for the two-layer Ising model. The average value for each K is calculated after its relaxation time. (data are the results for the lattice that each layer has 2500×2500 sites, starting from the homogeneous initial state with all +1, time steps = 50000)

of such calculation. As it is seen, before critical point ($K<K_C$), $<m>=0$ and after that ($K> K_C$), $<m> \neq 0$. The obtained values of the critical point from this approach is K_C =0.310 for the two-layer Ising model.

For calculation of χ for each K, first we have calculated the value of $(m-<m>)^2$ in each time step. Then these values are averaged in a same way explained above. According to eq. 8 this average could be used for computation of χ. Using eq. 9 for calculation of the specific heat (C), we have done it in a same way described above. Figures 5 and 6 show the graphs of χ vs. K and C vs. K, respectively, for the two-layer Ising model. These graphs are the other ways for obtaining the critical point. The maximum of these graphs indicates the critical point. The obtained value for K_C from these graphs is 0.310 for the two-layer Ising model.

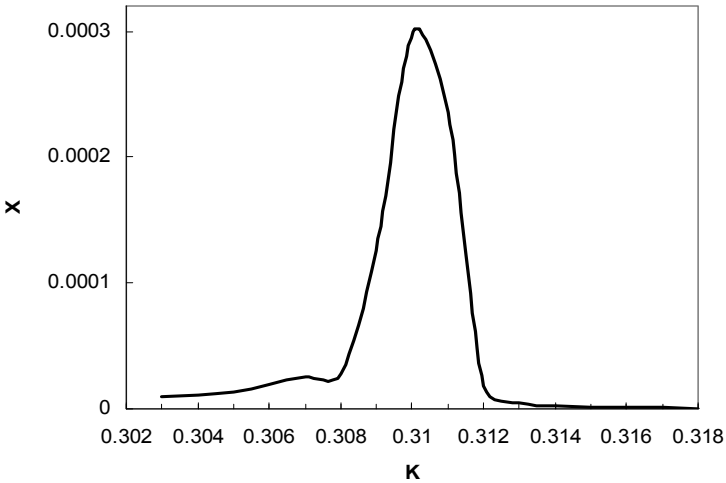


Fig. 5. Magnetization susceptibility per spin (χ) versus K for the two-layer Ising model. (The calculated data are the results for the lattice for which each layer has 2500×2500 sites, starting from the homogeneous initial state with all spins up, time steps = 50000)

3. Isotropic two-layer 3-state Potts model

Although we do not know the exact solution of the Potts model for any two-layer at present time, a large amount of numerical information has been accumulated for the critical properties of the various Potts models. Consider a two-layer square lattice with the periodic boundary condition, each layer with p columns and r rows. Each layer has then $r \times p$ sites and the number of the sites in the lattice is $2 \times r \times p = N$. We consider the next nearest neighbor interactions as well, so the number of neighbors for each site is 5. For any site we define a spin variable $\sigma^{1(2)}(i,j) = 0, \pm 1$ so that $i = 1,...r$ and $j = 1,...,p$. The configurational energy of a standard 3-state Potts model is given (Asgari et al., 2004) as:

$$\frac{E(\sigma)}{kT} = \sum_{i=1}^{r,*} \sum_{j=1}^{p,*} \sum_{n=1}^{2} -\{K_x \delta_{\sigma^n(i,j),\sigma^n(i+1,j)} + K_y \delta_{\sigma^n(i,j),\sigma^n(i,j+1)} + K_z \delta_{\sigma^1(i,j),\sigma^2(i,j)}\} \tag{14}$$

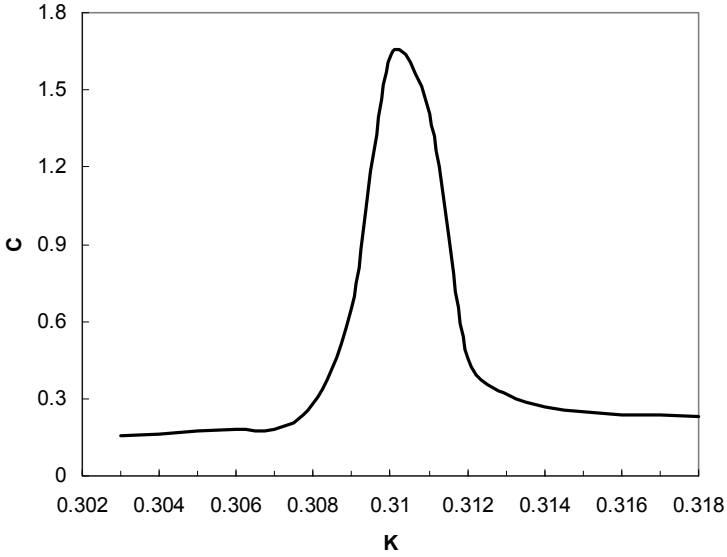


Fig. 6. Specific Heat per spin (*C*) versus *K* for the two-layer Ising model. (The calculated data are the results for the lattice for which each layer has 2500×2500 sites, starting from the homogeneous initial state with all spins up, time steps = 50000)

Where

$$\delta_{i,j} = 1 \text{ for } i = j$$
$$\delta_{i,j} = 0 \text{ for } i \neq j \tag{15}$$

and * indicates the periodic boundary condition and K_x and K_y are the nearest-neighbor interactions within each layer in x and y directions, respectively, and K_z is the interlayer coupling. Therefore, the configurational energy per spin is

$$e = \frac{E(\sigma)}{kTN} \tag{16}$$

and the general equations (5-9) are applicable to this model. For quantitative computation of the critical temperature of a two-layer 3-state Potts model, we considered the isotropic ferromagnetic case which $K_x = K_y = K_z \geq 0$. We have used a two-layer square lattice that each layer has 1500×1500 sites and to reduce the finite size effects the periodic boundary condition is used. Each site can have a value of +1, -1 or zero. We used the Glauber method with checkerboard approach to update the sites. Namely, each layer is like a checkered surfaces and at first, the updating is done for the white parts of the first layer. Then the black ones are updated. After which, this approach is done for the second layer. The updating of +1 spins is based on the probabilistic rules. The probability that spin of one site will be +1 (p_i^+) is given by

$$p_i^+ = \frac{e^{-\beta E_i^+}}{e^{-\beta E_i^+} + e^{-\beta E_i^-} + e^{-\beta E_i^0}} \tag{17}$$

Hence, probability that a given spin to be -1 (p_i^-) is

$$p_i^- = \frac{e^{-\beta E_i^-}}{e^{-\beta E_i^+} + e^{-\beta E_i^-} + e^{-\beta E_i^0}} \tag{18}$$

and for the zero state we have,

$$p_i^0 = 1 - (p_i^+ + p_i^-) \tag{19}$$

where

$$\begin{aligned} E_i^{\pm,0} = & -K_x\{\delta_{\sigma^n(i,j),\sigma^n(i+1,j)} + \delta_{\sigma^n(i,j),\sigma^n(i-1,j)}\} \\ & -K_y\left\{\delta_{\sigma^n(i,j),\sigma^n(i,j+1)} + \delta_{\sigma^n(i,j),\sigma^n(i,j-1)}\right\} \\ & -K_z\left\{\delta_{\sigma^n(i,j),\sigma^{n'}(i,j)}\right\} \end{aligned} \tag{20}$$

and

$$\begin{aligned} \sigma^n(i,j) &= +1 \text{ for } E_i^+ \\ \sigma^n(i,j) &= -1 \text{ for } E_i^- \\ \sigma^n(i,j) &= 0 \text{ for } E_i^0 \end{aligned} \tag{21}$$

It should be mentioned that in our approach, first we construct the probability matrix according to Eqs. (17-20) for different states of a cell in such a way that for each state it is sufficient to refer to the probability matrix and use the proper value of the probability. This leads to prevent similar calculations.

When we start the CA with the homogeneous initial state (namely, all sites have spin up or +1), before the critical point (K_C), the magnetization per spin (m) will decay rapidly to zero and fluctuate around that point. After the critical point, m will approach to the nonzero point and fluctuate around it and with increasing of K, the magnetization per spin will go toward its initial state (i.e. m = +1). But at the critical point, m will decay very slowly to zero with a great fluctuation. For each value of K, the time that m reaches to a special value and starts to fluctuate, is called the relaxation time (τ). On the other hand, the relaxation time is the time that system is thermalized. The value of τ can be obtained from the graph of m versus t. So one can see from this graph that the relaxation time increases before the critical point and reach its maximum at K_C, but after the critical point, τ decreases. So, in the critical point, the system last for long time to stabilize. Hence, the critical point could be obtained from the graph of τ versus K (Fig. 7).

Another way to get the critical point is the usage of the thermodynamic quantities after thermalization of the lattice. In another word, first we let the system to reach to a stable state after some time step ($t = \tau$). Next we let the system to be updated to the end of the automata

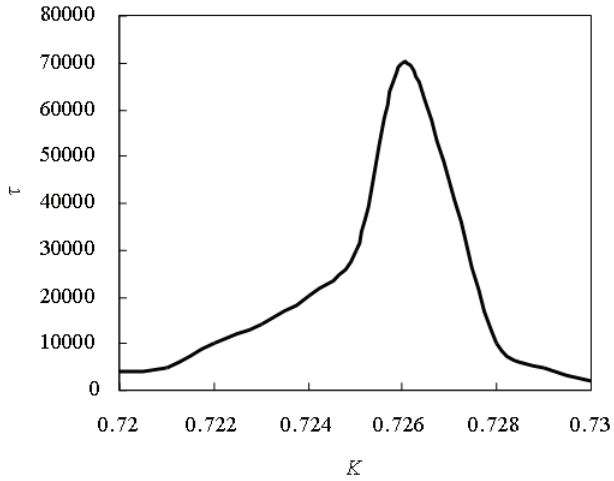


Fig. 7. Relaxation time (τ) versus coupling coefficients (K). (calculated data are the results of the lattice with 1500×1500 sites in each layer, start from homogeneous initial state with all of the spins up, time steps = 100000)

(t = 100000). For example, to calculate the average value of magnetization per spin (<m>), one should add all of values for m from the relaxation time to the end of the automata (or end of the time step) and divide the result to the numbers of steps. By drawing the graph of <m> versus K, we could get K_C. In this graph, for $K < K_C$, the value of <m> lies around zero. But it becomes nonzero at $K=K_C$, after which, its value increases gradually. For calculation of the susceptibility per spin χ (eq. 8), for each K, first we calculated the value of $(m - < m >)^2$ in each time step. Then these values are averaged by the same method explained above. Also the calculation of the specific heat C (eq. 9), may be done by a similar way. The graphs of χ versus K and C versus K, are another approach to obtain the critical point. The maximum of such graphs gives the critical point.

The result of such calculations are shown in figures 8-10 for the simplest case of the two-layer 3-state Potts model when $K_x=K_y=K_z=K \geq 0$. The obtained value for the critical point is 0.726 for this case.

4. Constructing the critical curve for anisotropic two-layer models

The previous approach could easily be used for calculation critical point of anisotropic two-layer Ising and Potts models which have different interlayer coupling coefficients ($K_x \neq K_y \neq K_z$) (Asgari & Ghaemi, 2006 and Asgari & Ghaemi, 2008). The critical points that are obtained for the two-layer Ising model in the case of different values of $\xi = K_z / K_x$ and $\sigma = K_y / K_x$ are given in Table 1.

The results are compared with other numerical methods and it is shown that they are in good agreement. So, this comparison confirms the reliability of our approach. In the next step, we have fitted the obtained results which are in Table 1 in order to get a general ansatz equation for the critical point for the anisotropic two-layer Ising model in terms. The results are compared with other numerical methods and it is shown that they are in good

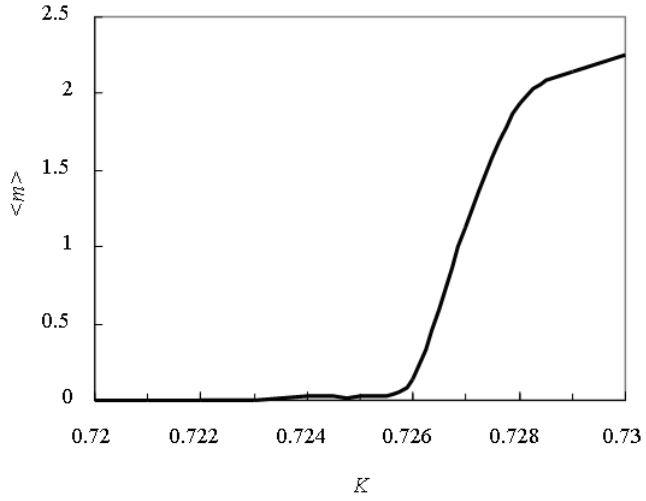


Fig. 8. <*m*> versus coupling coefficients (*K*). (calculated data are the results of the lattice with 1500 × 1500 sites in each layer, start from homogeneous initial state with all of the spins up, time steps = 100000)

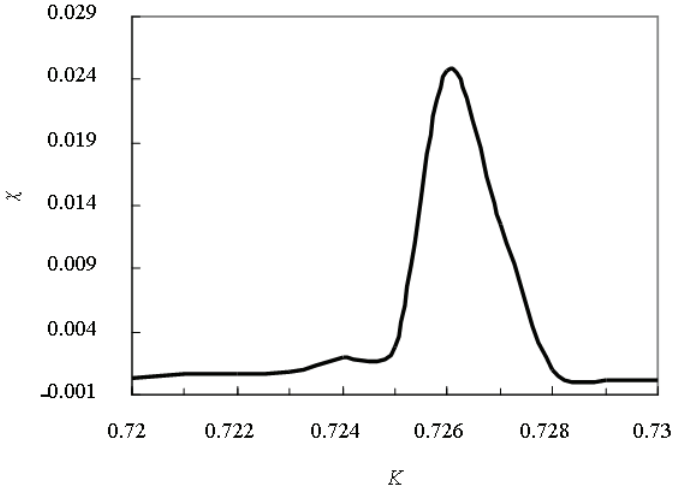


Fig. 9. Magnetization susceptibility per spin (χ) versus *K*. (calculated data are the results of the lattice with 1500 × 1500 sites in each layer, start from homogeneous initial state with all of the spins up, time steps = 100000)

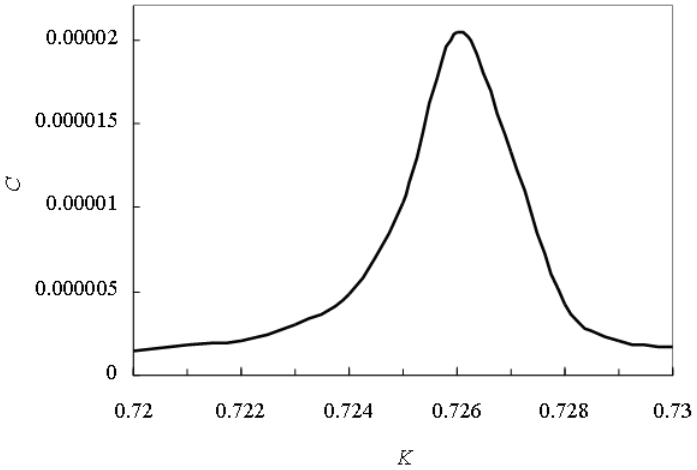


Fig. 10. Specific Heat per spin (C) versus K. (calculated data are the results of the lattice with 1500×1500 sites in each layer, start from homogeneous initial state with all of the spins up, time steps = 100000)

agreement. So, this comparison confirms the reliability of our approach. Furthermore, it is clear from the graph of M versus t (Fig. 1) that the critical point could be easily obtained with high precision using the CA approach. In the next step, we have fitted the obtained results which are in Table 1 in order to get a general ansatz equation for the critical point for the anisotropic two-layer Ising model in terms of inter- and intra-layer interactions (ξ and σ) as,

$$K_C^{-1} = a + b\xi^{1/2} + c\sigma^{1/2} \tag{22}$$

where, a = 0.170937(±0.002708), b = 0.724762(±0.003046), and c = 2.19985(±0.004167). This equation could be a reliable way to investigate the critical point for the anisotropic ferromagnetic two-layer Ising model by considering the nearest-neighbor interactions within each layer in the x and y directions and also the inter-layer coupling. So, having the desired values of ξ and σ, one could obtain the critical point for this model with acceptable precision. In the next step, we have done similar calculation to obtain the critical points for the anisotropic two-layer Potts model which is shown in Table 2.

Some of the results are compared with the recent transfer matrix method and it is shown that they are in good agreement. This comparison shows the reliability of our approach. The results are obtained from the graph of M versus t with high precision (see for example Fig. 2). Then, the obtained results have been fitted in order to obtain a general ansatz equation for the critical point for the anisotropic two-layer Potts model in terms of inter- and intra-layer couplings (ξ and σ) as follow,

$$K_C^{-1} = a + b\xi^{1/2} + c\sigma^{1/2} \tag{23}$$

where a = 0.162203(±0.001257), b = 0.246394(±0.003882), and c = 0.800764(±0.003190). This equation seems to be a useful expression in order to calculate the critical point for the

anisotropic two-layer Potts model in the lack of a general equation in terms of the nearest-neighbor interactions within each layer in the x and y directions and also the interlayer coupling.

K_z / K_x \ K_y / K_x	0.1	0.4	0.7	1.0	1.3
0.1	0.896	0.583	0.465	0.395	0.349
	0.879 a	0.582 a	0.464 a	0.397 a 0.3977 b	0.348 a
0.4	0.747	0.503	0.408	0.352	0.312
	0.763 a	0.510 a	0.413 a	0.354 a 0.3541 b	0.315 a
0.7	0.678	0.463	0.380	0.328	0.291
	0.686 a	0.465 a	0.381 a	0.330 a	0.293 a
1.0	0.639	0.436	0.357	0.310	0.276
	0.651 a	0.436 a	0.359 a	0.311 a 0.3117 b	0.277 a
1.3	0.608	0.414	0.341	0.296	0.265
	0.629 a	0.417 a	0.343 a	0.298 a	0.267 a

[a] From the transfer matrix method (Ghaemi et al., 2003).
[b] From the corner transfer matrix renormalization group method (Li et al., 2001)

Table 1. The critical points for the two-layer Ising model.

K_z / K_x \ K_y / K_x	0.001	0.01	0.1	0.5	1.0
0.001	5.23	4.08	2.43	1.37	1.00
0.01	4.60	3.73	2.35	1.35	0.98 0.974860[a]
0.1	3.48	3.00	2.06	1.25	0.92 0.902499[a]
0.5	2.85	2.34	1.64	1.05	0.80 0.795385[a]
1.0	2.47	2.14	1.44	0.94	0.72 0.726306[a]

[a] From the transfer matrix method (Mardani et al., 2005)

Table 2. The critical points for the 3-states two-layer Potts model

There are some features which can be mentioned here with the physical aspects according to eqs.22-23. It is shown that the critical point is proportional to ξ and σ in such a way that it increases when the value of ξ or σ decreases. As we have mentioned in earlier work (Asgari & Ghaemi, 2006), it is possible to increase the precision of the calculation by increasing the number of lattice size in order to make the system to have less fluctuation and so, determination of the critical point will be easier. Also, it should be noted that the number of

time steps should be high enough to determine the critical point especially in the case of fourth and more digits after the decimal point. However, it is clear that increasing the number of time steps and lattice size lead to decreasing the program rate. One way is to tabulate the probabilities in eqs. (17-20) and refer to such a table for each update and find the desire values for different probabilities in order to decrease the computational time. Also, parallel processing on cluster computers for the case of a large lattice size is another way to increase the program rate. The advantage of defined approach for the calculation of the critical point using the probabilistic CA is the possibility to get digits after the decimal point like fourth and more digits with higher precision.
Finally, it should be considered that it is possible to extend this approach to other lattice models such as triangular, hexagonal, and also other models like multi-states two-layer Potts model, 3-D Ising model, and asymmetric cases in order to obtain a general equation in the lack of the exact solution.

5. Calculation of the shift exponent

A shift exponent (φ) describes the deviation of the critical temperature from the critical temperature for the decoupled limit (Kz = 0). Some scaling theories were constructed to obtain the shift exponent (Oitmaa & Enting, 1975; Lipowski, 1998; Horiguchi & Tsushima 1997; Abe, 1970; and Suzuki, 1971). It was shown that for the two-layer model there is a relation between the critical point and the shift exponent as follow

$$T_C(\xi) - T_C(0) \propto \xi^{1/\varphi} \tag{24}$$

These theories predict that when the intra-layer interactions are the same in each layer, then $\varphi = \gamma$, where γ is the critical exponent describing divergence of susceptibility upon approaching the critical point (Abe, 1970; and Suzuki, 1971). Also it was mentioned that

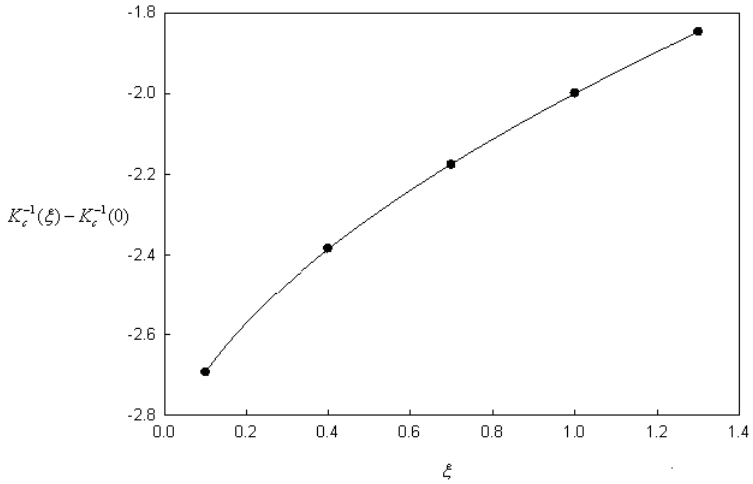


Fig. 10. The plot of $K_C^{-1}(\xi) - K_C^{-1}(0)$ versus ξ for the two-layer Ising model in the case of the equal intra-layer interactions (K_x = K_y)

when the intra-layer couplings changes in each lattice, then $\varphi = \gamma/2$. There has been several attempts to check these theories according to which some model gives good agreement and others show significant deviations (Lipowski, 1998 and Abe, 1970).
First, we have estimated the shift exponent for the two-layer Ising model in the case of equal intra-layer interactions. It was done considering following relation and the data of Table 1,

$$K_C^{-1}(\xi) - K_C^{-1}(0) \propto \xi^{1/\varphi} \tag{25}$$

Fig. 10. shows the plot of the left hand side of the relation (25) versus ξ. Then the results were fitted with a power equation and it was found that in this case the value of φ is 1.756(±0.0078) which is in good agreement with the other works (Lipowski, 1998).
This result is also in agreement with the arguments that the two-layer Ising model is in the same universality class as the two-dimensional Ising model with $\varphi = \gamma$ =1.75. Thereafter, we extend this calculation to the case of different intra-layer interactions. The obtained results for the shift exponent are shown in Table 3 which considerably different from those predicted by others. At present we cannot say in which point these scaling arguments are wrong but clearly they require reconsideration.

σ	Φ
0.1	2.807
0.4	2.066
0.7	1.906
1.0	1.756

Table 3. The shift exponent for the two-layar Ising Model (the exact value for K_C is 0.440687).

However, the results could be fitted into a rational ansatz equation in terms of intra-layer interactions (σ) as

$$\varphi = \frac{a_0 + \sigma}{a_1 + a_2\sigma} \tag{26}$$

where $\sigma = K_y / K_x$ and the universal coefficients are a_0=0.1803(±0.001349), a1= 0.0399(±0.000451), and a2 = 0.5995(±0.003601). As shown in Fig. 11, eq. (26) has a decay form and covers all calculated data for $\sigma \leq 1$.
In this step we have calculated the shift exponent for the two-layer Potts model. In the case of equal intra-layer interactions we have used the relation (25) and the data of Table 2 in order to calculate the shift exponent. The value for the shift exponent after fitting with a power equation is 1.582(±0.0128) which differs from the obtained value for the two-layer Ising model. This result shows that the two-layer Potts model is not in the same universality class as the two-dimensional and two-layer Ising model. We have done the calculation for different values of the intra-layer couplings. The results are shown in Table 4.
These results could be fitted into a rational ansatz equation in terms of the intra-layer couplings (σ) as

$$\varphi = \frac{b_0 + \sigma}{b_1 + b_2\sigma} \tag{27}$$

where $\sigma = K_y / K_x$ and the universal coefficients are b_0=0.0630(±0.000166), b_1=0.0181(±0.000103), and b_2= 0.6547(±0.002691). As shown in Fig. 12, eq. (27) has also a decay form and covers all calculated data for $\sigma \leq 1$.

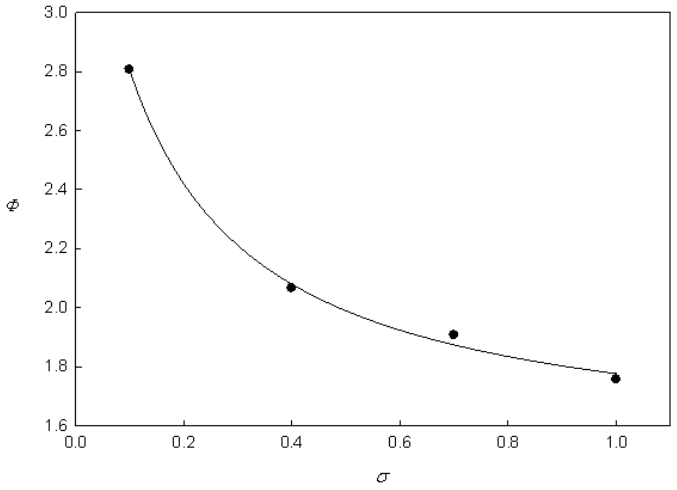


Fig. 11. The Plot of φ versus σ for the two-layer Ising model

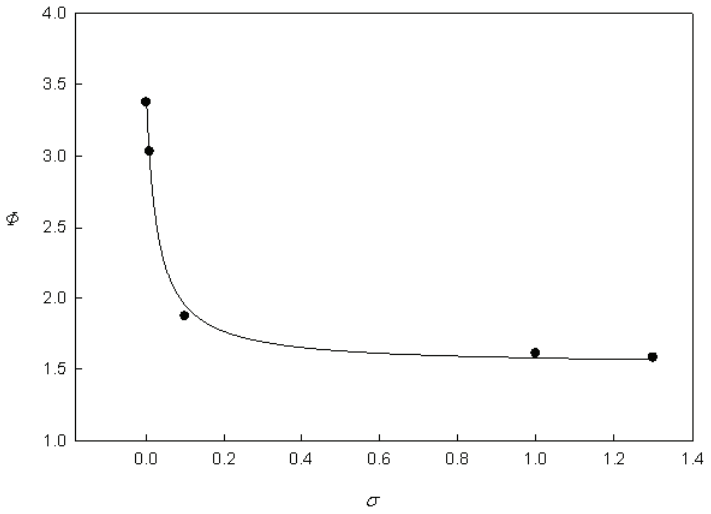


Fig. 12. The Plot of φ versus σ for the 3-states two-layer Potts model

σ	Φ
0.001	3.376
0.01	3.031
0.1	1.871
0.5	1.612
1.0	1.582

Table 4. The shift exponent for the 3-states two-layer Potts model the exact value for K_C is 1.005052).

6. Reference

Abe, R. (1970). *Prog. Theor. Phys.* 44 339.

Aktekin, N. (1999). *Annal Review of computational Physics* VII. Edited by Staufer, D., World Scientific Publishing Company 1-23

Allan, G.A.T. (1970). *Phys. Rev. B.* 1, 352

Angelini, L., Carappo, D., Pellicoro, M., Villani, M. (1995). *Physica A.* 19, 447

Angelini, L., Carappo, D., Pellicoro, M., Villani, M. (1997). *Physica A.* 237, 320

Asgari, Y., Ghaemi, M., Mahjani, M.G. (2004) *Lecture Notes in Computer Science.*, 3305, 709-718

Asgari, Y.; Ghaemi M.; (2006). *J. Theo. Comp. Chem.*, 2, 141-150

Asgari, Y.; Ghaemi M.; (2008). *Physica A*, , 387, 1937–1946

Ballentine, L.E. (1964). *Physica*, 30, 1231

Binder, K. (1974). *Thin Solid Films.* 20, 367

Creutz, M. (1986). *Annals of physics.* 167, 62-76

Domany, E., Kinzel, W. (1984). *Phys. Rev. Let.* 53, 4, 311-314

Ghaemi, M., Ghannadi, M., Mirza, B. (2003). *J. Phys. Chem. B.* 107, 829-831

Ghaemi, M., Mirza, B., Parsafar, G.A., (2004). *J. Theor. & Comp. Chem.* 3, 217-224.

Glauber, R.J.: J. Math. Phys. (1963) 4, 294

Glotzer, S.C., Stauffer, D., Sastry, S. (1990). *Physica A* 164 1-11.

Herrmann, H. J. (1986). *J. Stat. Phys.* 45, 145

Hilfer, R., Biswal, B., Mattutis, H.G., Janke, W. (2003). *Phys. Rev. E* 68 046123.

Horiguchi, T., Lipowski, A., Tsushima, N. (1996). *Physica A.* 224, 626

Horiguchi, T. and Tsushima, N. (1997). *Physica A* 238

Huang, K. (1987). *Statistical mechanics.* John Wiley and Sons, 2nd Edition,

Janke, W. (1998). *Physica A* 254 164-178.

Jan, N.(1990). *J. Physique.* 51, 201

Kinzel, W. (1985). *Z. Phys. B.* 58, 229-244

Kremer, S. and Wolf, D.E. (1992). *Physica A* 182 542-556.

Li, Z.B., Shuai, Z., Wang, Q., Luo, H.J., Schulke, L. (2001). *J. Phys. A.* 34, 6069-6079

Lipowski, A. (1998). *Physica A* 250 373-383.

Lipowski, A., Suzuki, M. (1998). *Physica A.* 250, 373

MacIsaac, A.B. (1990) *J. Phys. A* 23 899-903.

Mardani, T. , Mirza, B., Ghaemi, M. (2005). *Phys. Rev. E* 72 026127.

Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E. (1953). *J. Chem. Phys.* 21, 1087

Moukarzel, C. and Parga, N. (1989). *J. Phys. A* 22 943.

Newman, M.E., Barkema, G.T. (2001). *Monte Carlo Methods in Statistical Physics*. Oxford University Press Inc., New York, Reprinted. Chap. 3-4.
Oitmaa, J., Enting, G. (1975). *J. Phys. A*. 8, 1097
Onsager, L. (1944). *Phys. Rev*. 65, 117
Pomeau, Y. (1984). *J. Phys. A*. 17, 415
Stauffer, D. (2000) . *Comp. Phys. Comm*. 127 113-119.
Stauffer, D. (1997) . *Int. J. Mod. Phys. C* 8 6, 1263-1266.
Suzuki, M. (1971). *Prog. Theor. Phys.* 46 1054.
Toffoli, T., Margolus, N. (1990). *Physica D*. 45, 229-253
Vichniac, G. (1984). *Physica D*10. 96-115
Wosiek, J. (1994). *Phys. Rev. B*. 49, 15023
Wu, F.Y. (1982). *Rev. Mod. Phys.* 54, 235
Zabolitzky, J. G. and Herrmann, H. J. (1988). *J. Comp. Phys.* 76 426-447.
Zheng, B. (1998). *Int. J. Mod. Phys. B* 12 14, 1419-1484.

22

Propositional Proof Complexity and Cellular Automata

Stefano Cavagnetto
School of Computing and Prague College Research Centre, Prague College
Polská 10, Prague
Czech Republic

1. Introduction

Two fields connected with computers, automated theorem proving on one side and computational complexity theory on the other side, gave the birth to the field of propositional proof complexity in the late '60s and '70s. In this chapter we consider how classic propositional logic and in particular Propositional Proof Complexity can be combined with the study of Cellular Automata. It is organized as follows: in the next section we recall some of the basic definitions in computational complexity theory [1]. In the same section we introduce some basic definitions from propositional proof complexity[2] and we recall an important result by Cook and Reckhow (20) which gives an interesting link between complexity of propositional proofs and one of the most beautiful open problem in contemporary mathematics[3]. Section 3 deals with cellular automata and on how Propositional Logic and techniques from Propositional Proof Complexity can be employed in order to give a new proof of a famous theorem in the field known as Richardson's Theorem[4]. In the chapter some complexity results regarding Cellular Automata are considered and described. The ending section of the chapter deals with a new proof system based on cellular automata and also it outlines some of the open problems related to it.

2. Some technical preliminaries

In 1936 Alan Turing (63) introduced the standard computer model in computability theory, the Turing machine. A Turing machine M consists of a finite state control (a finite program) attached to read/write head which moves on an infinite tape. The tape is divided into squares. Each square is capable of storing one symbol from a finite alphabet Γ. $b \in \Gamma$, where b is the blank symbol. Each machine has a specified input alphabet $\Sigma \subseteq \Gamma$ where $b \notin \Sigma$. M is in some finite state q (in a specified finite set Q of possible states), at each step in a computation. At the beginning a finite input string over Σ is written on adjacent squares of the tape and all

[1] For a self-contained exposition of the field the interested reader can see (56), (49).
[2] There are many survey papers on propositional proof complexity offering different emphasis; the interested reader can see (64), (17)and (51).
[3] The famous $\mathcal{P}$ versus $\mathcal{NP}$ problem, (22), (50), (57), (65), (58).). In this chapter, the next section regarding computational complexity follows in detail Cook's paper (22)
[4] This new proof was given by the present author in (14); section 3 follows in detail this work.

other squares are blank. The head scans the left-most symbol of the input string, and M is in the initial state q_0. At every step M is in some state q and the head is scanning a square on the tape containing some symbol s, and the action performed depends on the pair (q, s) and is specified by the machine's transaction function (or program) δ. The action consists of printing a symbol on the scanned square, moving the head left or right of one square, and taking a new state.
Formally the model introduced by Turing can be presented as follows. It is a tuple $\langle \Sigma, \Gamma, Q, \delta \rangle$ where Σ, Γ, Q are nonempty sets with $\Sigma \subseteq \Gamma$ and $b \in \Gamma - \Sigma$. The state set Q contains three special states q_0, q_{accept} and q_{reject}. The transition function δ satisfies:

$$\delta : (Q - \{q_{accept}, q_{reject}\}) \times \Gamma \to Q \times \Gamma \times \{-1, 1\}.$$

$\delta(q, s) = (q', s', h)$ is interpreted as: if M is in the state q scanning the symbol s then q' is the new state, s' is the new symbol printed on the tape, and the tape head moves left or right of one square (this depends whether h is -1 or 1). We assume $Q \cap \Gamma = \emptyset$. A configuration of M is a string xqy with $x, y \in \Gamma^*$, y is not the empty string, $q \in Q$. We interpret the configuration xqy as follows: M is in state q with xy on its tape, with its head scanning the left-most symbol of y.

Definition 2.1. *If C and C' are configurations, then $C \overset{M}{\to} C'$ if $C = xqsy$ and $\delta(q, s) = (q', s', h)$ and one of the following holds:*

1. *$C' = xs'q'y$ and $h = 1$ and y is nonempty.*
2. *$C' = xs'q'b$ and $h = 1$ and y is nonempty.*
3. *$C' = x'q'as'y$ and $h = -1$ and $x = x'a$ for some $a \in \Gamma$.*
4. *$C' = q'bs'y$ and $h = -1$ and x is empty.*

A configuration xqy is halting if $q \in \{q_{accept}, q_{reject}\}$.

Definition 2.2. *A computation of M on input $w \in \Sigma^*$, where Σ^* is the set of all finite string over Σ, is the unique sequence $C_0, C_1, \ldots$ of configurations such that $C_0 = q_0 w$ (or $C_0 = q_0 b$ if w is empty) and $C_i \overset{M}{\to} C_{i+1}$ for each i with C_{i+1} in the computation, and either the sequence is infinite or it ends in a halting configuration.*

If the computation is finite, then the number of steps is one less than the number of configurations; otherwise the number of steps is infinite.

Definition 2.3. *M accepts w if and only if the computation is finite and the final configuration contains the state q_{accept}.*

The elements belonging to the class $\mathcal{P}$ are languages. Let Σ be a finite alphabet with at least two elements, and Σ^*, as above, the set of all finite strings over Σ. A language over Σ is $L \subseteq \Sigma^*$. Each Turing machine M has an associated input alphabet Σ. For each string $w \in \Sigma^*$ there exists a computation associated with M and with input w. We said above[5] that M accepts w if this computation terminates in the accepting state.[6] The language accepted by M that we denote by $L(M)$ has associated alphabet Σ and is defined by

[5] See Definition 2.3.
[6] Notice that M fails to accept w if this computation ends in the rejecting state, or if the computation fails to terminate.

$$L(M) = \{w \in \Sigma^* |\ M \text{ accepts } w\}.$$

Let $t_M(w)$ be the number of steps in the computation of M on input w. If this computation never halts then $t_M(w) = \infty$. For $n \in \mathbb{N}$ we denote by $T_M(n)$ the worst case run time of M; i.e.

$$T_M(n) = max\{t_M(w)|\ w \in \Sigma^n\}$$

where Σ^n is the set of all strings over Σ of length n. Thus, we say that M runs in polynomial time if there exists k such that for all n, $T_M(n) \leq n^k + k$. Then the class $\mathcal{P}$ of languages can be defined by the condition that a language L is in $\mathcal{P}$ if $L = L(M)$ for some Turing machine M which runs in polynomial time.

The complexity class $\mathcal{NP}$ can be defined as follows using the notion of a checking relation, which is a binary relation $R \subseteq \Sigma^* \times \Sigma_1^*$ for some finite alphabets Σ and Σ_1. We associate with each such relation R a language L_R over $\Sigma \cup \Sigma_1 \cup \{\#\}$ defined by $L_R = \{w\#y|\ R(w, y)\}$, where the symbol $\# \notin \Sigma$. R is polynomial time if and only if $L_R \in \mathcal{P}$. The class $\mathcal{NP}$ of languages can be defined by the condition that a language L over Σ is in $\mathcal{NP}$ if there is $k \in \mathbb{N}$ and a polynomial time checking relation R such that for all $w \in \Sigma^*$,

$$w \in L \Longleftrightarrow \exists y(|y|\ \leq |w|^k \wedge R(w, y))$$

where $|w|$ and $|y|$ denote the lengths of w and y, respectively.

The question of whether $\mathcal{P} = \mathcal{NP}$ is one of the greatest unsolved problem in theoretical computer science and in contemporary mathematics. Most researchers believe that the two classes are not equal (of course, it is easy to see that $\mathcal{P} \subseteq \mathcal{NP}$). At the beginning of the '70s Cook and Levin, independently, pointed out that the individual complexity of certain problems in $\mathcal{NP}$ is related to that of the entire class. If a polynomial time algorithm exists for any of these problems then all problems in $\mathcal{NP}$ would be polynomially solvable. These problems are called $\mathcal{NP}$-complete problems. Since that time thousands of $\mathcal{NP}$-complete problems have been discovered. We recall here only the first and probably one of the most famous of them, the satisfiability problem. For a collection of these problems the interested reader can see (29).

Let ϕ be a Boolean formula in the De Morgan language with constants 0, 1 (the truth values $FALSE$ and $TRUE$) and propositional connectives: unary $\neg$ (the negation) and binary $\wedge$ and $\vee$ (the conjunction and the disjunction, respectively). A Boolean formula is said to be satisfiable if some assignment of 0s and 1s to the variables makes the formula evaluate to 1. The satisfiability problem is to test whether a Boolean formula ϕ is satisfiable; this problem is denoted by SAT. Let $SAT = \{\langle\phi\rangle\ |\phi \text{ is a satisfiable Boolean formula}\}$.

Theorem 2.4 (Cook (19), Levin (42)). *$SAT \in \mathcal{P}$ if and only if $\mathcal{P} = \mathcal{NP}$.*

Suppose that L_i is a language over Σ_i, $i = 1, 2$. Then $L_1 \leq_p L_2$ (L_1 is polynomially reducible to L_2) if and only if there is a polynomial time computable function $f : \Sigma_1^* \to \Sigma_2^*$ such that

$$x \in L_1 \Longleftrightarrow f(x) \in L_2,$$

for all $x \in \Sigma_1^*$.

Definition 2.5. *A language L is $\mathcal{NP}$-complete if $L \in \mathcal{NP}$ and every language $L' \in \mathcal{NP}$ is polynomial time reducible to L.*

A language L is said $\mathcal{NP}$-hard if all languages in $\mathcal{NP}$ are polynomial time reducible to it, even though it may not be in $\mathcal{NP}$ itself.
The heart of Theorem 2.4 is the following one.

Theorem 2.6. *SAT is $\mathcal{NP}$-complete.*

Consider the complement of SAT. Verifying that something is not present seems more difficult than verifying that it is present, thus it seems not obviously a member of $\mathcal{NP}$. There is a special complexity class, $co\mathcal{NP}$, containing the languages that are complements of languages of $\mathcal{NP}$. This new class leads to another open problem in computational complexity theory. The problem is the following: is $co\mathcal{NP}$ different from $\mathcal{NP}$? Intuitively the answer to this problem, as in the case of the $\mathcal{P}$ versus $\mathcal{NP}$ problem, is positive. But again we do not have a proof of this.
Notice that the complexity class $\mathcal{P}$ is closed under complementation. It follows that if $\mathcal{P} = \mathcal{NP}$ then $\mathcal{NP} = co\mathcal{NP}$. Since we believe that $\mathcal{P} \neq \mathcal{NP}$ the previous implication suggests that we might attack the problem by trying to prove that the class $\mathcal{NP}$ is different from its complement. In the next section we will see that this is deeply connected with the study of the complexity of propositional proofs in mathematical logic.
We conclude this introductory section by recalling some basic definitions from circuit complexity which will be used afterwards and the classical notation for the estimate of the running time of algorithms, the so called Big-O and Small-o notation for time complexity.

Definition 2.7. *A Boolean Circuit C with n inputs variables $x_1, \ldots, x_n$ and m outputs variables $y_1, \ldots, x_m$ and basis of connectives $\Omega = \{g_1, \ldots, g_k\}$ is a labelled acyclic directed graph whose out-degree 0 nodes are labelled by y_j's, in-degree 0 nodes are labeled by x_i's or by constants from Ω, and whose in-degree $\ell \geq 1$ nodes are labeled by functions from Ω of arity ℓ.*

The circuit computes a function $C : 2^n \to 2^m$ in an obvious way, where we identify $\{0,1\}^n = 2^n$.

Definition 2.8. *The size of a circuit is the number of its nodes. Circuit complexity $C(f)$ of a function $f : 2^n \to 2^m$ is the minimal size of a circuit computing f.*

In one form of estimation of the running time of algorithms, called the asymptotic analysis, we look for understanding the running time of the algorithm when large inputs are considered. In this case we consider just the highest order term of the expression of the running time, disregarding both coefficient of that term and any other lower term. Throughout this work we will use the asymptotic notation to give the estimate of the running time of algorithms and procedures. Thus we think that for a self-contained presentation it is perhaps worth to recall the Big-O and Small-o notation for time complexity. Let $\mathbb{R}^+$ be the set of real numbers greater than 0. Let f and g be two functions $f, g : \mathbb{N} \to \mathbb{R}^+$. Then $f(n) = O(g(n))$ if positive integers c and n_0 exist so that for every integer $n \geq n_0$, $f(n) \geq cg(n)$.[7] In other words, this definition points out that if $f(n) = O(g(n))$ then f is less than or equal to g if we do not consider differences up to a constant factor. The Big-O notation gives a way to say that one function is asymptotically no more than another. The Small-o gives a way to say that one function is asymptotically less than another. Formally, let f and g be two functions $f, g : \mathbb{N} \to \mathbb{R}^+$. Then $f(n) = o(g(n))$ if $\lim_{n\to\infty} f(n)/g(n) = 0$.

[7] When $f(n) = O(g(n))$ we say that $g(n)$ is an asymptotic upper bound for $f(n)$.

2.1 The complexity of propositional proofs

The complexity of propositional proofs has been investigated systematically since late '60s.[8] Cook and Reckhow in (20), (21) gave the general definition of propositional proof system. To be able to introduce their definition that plays a central role in our work and is foundamental in the theory of complexity of the propositional proofs, we start from an example that must be familiar to anyone who has some basic knowledge of mathematical logic.

Let $TAUT$ be the set of tautologies in the De Morgan language[9] with constants 0, 1 (the truth values $FALSE$ and $TRUE$) and propositional connectives: unary $\neg$ (the negation) and binary $\wedge$ and $\vee$ (the conjunction and the disjunction, respectively). The language also contains auxiliary symbols such as brackets and commas. The formulas are built up using the constants, the atoms (propositional variables) $p_0, \ldots, p_n$, and the connectives. Consider the following example of set of axioms taken from Hilbert's and Ackermann's work (31), where $A \to B$ is just the abbreviation of $\neg A \vee B$,

1. $A \vee (A \to A)$
2. $A \to (A \vee B)$
3. $(A \vee B) \to (B \vee A)$
4. $(B \to C) \to ((A \vee B) \to A \vee C))$

The only inference rule is *modus (ponendo) ponens*[10] (MP), $A \to B, A/B$ (i.e. $A, \neg A \vee B/B$).

The literature of mathematical logic contains a wide variety of propositional proof systems formalized with a finite number of axiom schemes and a finite number of inference rules. The example above is just one of many possible different formalizations. Any of such systems is called a *Frege System* and denoted by F. A more general definition for Frege systems can be given using the concept of a Frege rule.

Definition 2.9. *A Frege rule is a pair* $(\{\phi_1(p_0, ..., p_n), ..., \phi_k(p_0, ..., p_n)\}, \phi(p_0, ..., p_n))$, *such that the implication*

$$\phi_1 \wedge ... \wedge \phi_k \to \phi$$

is a tautology. We use $p_0, \ldots, p_n$ *for propositional variables and usually we write the rule as*

$$\frac{\phi_1, \ldots, \phi_k}{\phi}.$$

Notice that a Frege rule can have zero premises and in which case it is called an axiom schema (as the example above for the axioms (1) to (4)).

Definition 2.10. *A Frege system F is determined by a finite complete set of connectives and a finite set of Frege rules. A formula* ϕ *has a proof in F if and only if* $\phi \in TAUT$.[11] *F is implicationally complete.*[12]

As consequence of the schematic formalization we have that, the relation "w is a proof of ϕ in F" is a polynomial time relation of w and ϕ.

[8] The earliest paper on the subject is an article by Tseitin (62).

[9] Introduced in the previous section when we defined the problem SAT.

[10] In Latin, the mode that affirms by affirming.

[11] The "if" direction is the completeness and the "only" direction is the soundness of F.

[12] Recall that F is implicationally complete if and only if any ϕ can be proved in F from any set $\{\delta_1, \cdots, \delta_n\}$ if every truth assignment satisfying all δ_i's satisfies also ϕ.

We consider all finite objects in our proofs as encoded in the binary alphabet $\{0,1\}$. In particular, we consider $TAUT$ as a subset of $\{0,1\}^*$. The length of a formula ϕ is denoted $|\phi|$. The properties above lead to a more abstract definition of proof system (20),

Definition 2.11 (Cook Reckhow (20))**.** *A propositional proof system is any polynomial time computable function $P : \{0,1\}^* \to \{0,1\}^*$ such that $Rng(P) = TAUT$. Any $w \in \{0,1\}$ such that $P(w) = \phi$ is called a proof of ϕ in P.*

Any Frege system can be seen as a propositional proof system in this abstract perspective. In fact, consider the following function P_F,

$$P_F(w) = \begin{cases} \phi & \text{if } w \text{ is a proof of } \phi \text{ in } P \\ 1 & \text{otherwise} \end{cases}$$

Definition 2.12. *A propositional proof system P is polynomially bounded if there exists a polynomial $p(x)$ such that any $\phi \in TAUT$ has a proof w in P of size $|w| \leq p(|\phi|)$.*

In other words, any propositional proof system P that proves all tautologies in polynomial size is polynomially bounded. In (20) has been proved the following fundamental theorem relating propositional proof complexity to computational complexity theory. We report the theorem and the sketch of the proof.

Theorem 2.13 (Cook Reckhow (20))**.** *$\mathcal{NP} = co\mathcal{NP}$ if and only if there exists a polynomially bounded proof system P.*

Proof. Notice that since SAT is $\mathcal{NP}$-complete and for all $\neg\phi$, $\neg\phi \notin TAUT$ if and only if $\phi \in SAT$, $TAUT$ must be $co\mathcal{NP}$-complete. Assume $\mathcal{NP} = co\mathcal{NP}$. Then by hypothesis $TAUT \in \mathcal{NP}$. Hence there exists a polynomial $p(x)$ and a polynomial time relation R such that for all ϕ,

$$\phi \in TAUT \text{ if and only if } \exists y(R(\phi, y) \wedge |y| \leq p(|\phi|)).$$

Now define the propositional proof system as follows:

$$P(w) = \begin{cases} \phi & \text{if } \exists y(R(\phi, y) \text{ and } w = (\phi, y) \\ 1 & \text{otherwise} \end{cases}$$

It is clear that P is polynomially bounded.
For the opposite direction assume that P is a polynomially bounded propositional proof system for $TAUT$. Let $p(x)$ be a polynomial satisfying Definition 2.12. Since for all ϕ,

$$\phi \in TAUT \text{ if and only if } \exists w(P(w) = \phi \wedge |w| \leq p(|\phi|)),$$

we get that $TAUT \in \mathcal{NP}$. Let $R \in co\mathcal{NP}$. By the $co\mathcal{NP}$-completeness of $TAUT$, R is polynomially reducible to $TAUT$. Since $TAUT \in \mathcal{NP}$ then so is R. This shows that $co\mathcal{NP} \subseteq \mathcal{NP}$ and consequently also that $co\mathcal{NP} = \mathcal{NP}$.

□

Hence, if we believe that $\mathcal{NP} \neq co\mathcal{NP}$ then there is no polynomially bounded propositional proof system for classical tautologies. Recall from the previous section that if $\mathcal{NP} \neq co\mathcal{NP}$ then $\mathcal{P} \neq \mathcal{NP}$. To prove that $\mathcal{NP} \neq co\mathcal{NP}$ is equivalent, by Theorem 2.13, to prove that there is no propositional proof system that proves all classical tautologies in polynomial size.

This line of research gave rise to the program of proving lower bounds for many propositional proof systems. As mentioned in (38) it would be unlikely to prove that $\mathcal{NP} \neq co\mathcal{NP}$ in this incremental manner by showing exponential lower bounds for all the proof systems known.[13] This is like trying to prove a universal statement by proving all its instances. Despite that, we may hope to uncover some hidden computational aspect in these lower bounds and thus to reduce the conjecture to some intuitively more rudimentary one. For more discussion on this the reader can see (38).
We conclude this section with the notion of polynomial simulation introduced in (20). The definition 2.14 is simply a natural notion of quasi-ordering of propositional proof systems by their strength.

Definition 2.14. *Let P and Q be two propositional proof systems. The system P polynomially simulates Q, $P \geq_p Q$ in symbols, if and only if there is polynomial time computable function $g : \{0,1\}^* \to \{0,1\}^*$ such that for all $w \in \{0,1\}^*$, $P(g(w)) = Q(w)$.*

The function g translates proofs in Q into proofs in P of the same formula. Since in the definition above g is a polynomial time function, then the length of the proofs in P will be at most polynomially longer than the length of the original proofs in the system Q.

2.2 Resolution

The logical calculus Resolution R is a refutation system for formulas in conjunctive normal form. This calculus is popularly credited to Robinson (55) but it was already contained in Blake's thesis (9) and is an immediate consequence of Davis and Putnam work (26).
A literal ℓ is either a variable p or its negation $\bar{p}$. The basic object is a clause, that is a finite or empty set of literals, $C = \{\ell_1, \ldots, \ell_n\}$ and is interpreted as the disjunction $\bigvee_{i=1}^{n} \ell_i$. A truth assignment $\alpha : \{p_1, p_2, \ldots\} \to \{0,1\}$ satisfies a clause C if and only if it satisfies at least one literal l_i in C. It follows that no assignment satisfies the empty clause, which it is usually denoted by $\{\}$. A formula ϕ in conjunctive normal form is written as the collection $\mathcal{C} = \{C_1, \ldots, C_m\}$ of clauses, where each C_i corresponds to a conjunct of ϕ. The only inference rule is the resolution rule, which allows us to derive a new clause $C \cup D$ from two clauses $C \cup \{p\}$ and $D \cup \{\bar{p}\}$

$$\frac{C \cup \{p\}\ D \cup \{\bar{p}\}}{C \cup D}$$

where p is a propositional variable. C does not contain p (it may contain $\bar{p}$) and D does not contain $\bar{p}$ (it may contain p). The resolution rule is sound: if a truth assignment $\alpha : \{p_1, p_2, \ldots\} \to \{0,1\}$ satisfies both upper clauses of the rule then it also satisfies the lower clause.
A resolution refutation of ϕ is a sequence of clauses $\pi = D_1, \ldots, D_k$ where each D_i is either a clause from ϕ or is inferred from earlier clauses D_u, D_v, $u, v < i$ by the resolution rule and the last clause $D_k = \{\}$. Resolution is sound and complete refutation system; this means that a refutation does exist if and only if the formula ϕ is unsatisfiable.

Theorem 2.15. *A set of clauses $\mathcal{C}$ is unsatisfiable if and only if there is a resolution refutation of the set.*

Proof. The "only-if part" follows easily from the soundness of the resolution rule. Now, for the opposite direction, assume that $\mathcal{C}$ is unsatisfiable and such that only the literals p_1,

[13] Unless there is an optimal proof system.

$\neg p_1, \ldots, p_n, \neg p_n$ appear in $\mathcal{C}$. We prove by induction on n that for any such $\mathcal{C}$ there is a resolution refutation of $\mathcal{C}$.

Basis Case: If $n = 1$ there is nothing to prove: the set $\mathcal{C}$ must contain $\{p_1\}$ and $\{\neg p_1\}$ and then by the resolution rule we have $\{\}$.

Induction Step: Assume that $n > 1$. Partition $\mathcal{C}$ in four disjoint sets:

$$\mathcal{C}_{00} \cup \mathcal{C}_{01} \cup \mathcal{C}_{10} \cup \mathcal{C}_{11}$$

of those clauses which contain no p_n and no $\neg p_n$, no p_n but do contain $\neg p_n$, do contain p_n but not $\neg p_n$ and contain both p_n and $\neg p_n$, respectively. Produce a new set of clauses $\mathcal{C}'$ by:

(1) Delete all clauses from $\mathcal{C}_{11}$.

(2) Replace $\mathcal{C}_{01} \cup \mathcal{C}_{10}$ by the set of clauses that are obtained by the application of the resolution rule to all pairs of clauses $C_1 \cup \{\neg p_n\}$ from $\mathcal{C}_{01}$ and to $C_2 \cup \{p_n\}$ from $\mathcal{C}_{10}$.

The new set of clauses do not contain either p_n or $\neg p_n$. It is easy to see that the new set of clauses $\mathcal{C}'$ is also satisfiable. Any assignment $\alpha' : \{p_1, \ldots, p_{n-1}\} \to \{0, 1\}$ satisfies all clauses C_1 such that $C_1 \cup \{\neg p_n\} \in \mathcal{C}_{01}$, or all clauses C_2 such that $C_2 \cup \{p_n\} \in \mathcal{C}_{01}$. Hence α' can be extended to a truth assignment α satisfying $\mathcal{C}$, which is a contradiction because by our hypothesis $\mathcal{C}$ is unsatisfiable.

□

A resolution refutation $\pi = D_1, \ldots, D_k$ can be represented as a directed acyclic graph (dag-like) in which the clauses are the vertices, and if two clauses $C \cup \{p\}$ and $D \cup \{\bar{p}\}$ are resolved by the resolution rule, than there exists a direct edge going from each of the two clauses to the resolvent $C \cup D$. A resolution refutation $\pi = D_1, \ldots, D_k$ is tree-like if and only if each D_i is used at most once as a hypothesis of an inference in the proof. The underlying graph of π is a tree. The proof system allowing exactly tree-like proofs is called tree-like resolution and denoted by R^*.
In propositional proof complexity, perhaps the most important relation between dag-like refutations and refutations in R^* is that the former can produce exponentially shorter refutations then the latter. A simple remark on this is that in a tree-like proof anything which is needed more than once in the refutation must be derived again each time from the initial clauses. A superpolynomial separation between R^* and R was given in (64), and later by others in (16) and (32). Later on, in (10) has been presented a family of clauses for which R^* suffers an exponential blow-up with respect to R. For an improvement of the exponential separation the reader can see (6).

2.3 Interpolation and effective interpolation

A basic result in mathematical logic is the Craig interpolation theorem (24). The theorem says that whenever an implication $A \to B$ is valid then there exists a formula I, called an interpolant, which contains only those symbols of the language occurring in A and B and such that the two implications $A \to I$ and $I \to B$ are both valid formulas. Craig's interpolation theorem is a fundamental result in propositional logic and in predicate logic as well.[14]

[14] Throghout all this work by Craig interpolation's theorem we mean the propositional version of it.

Finding an interpolant for the implication is a problem of some relevance with respect to computational complexity theory. Mundici in (45) pointed out the following. Let U and V be two disjoints $\mathcal{NP}$-sets, subsets of $\{0,1\}^*$. By the proof of the $\mathcal{NP}$-completeness of satisfiability (19) there are sequences of propositional formulas $A_n(p_1,\ldots,p_n,q_1,\ldots,q_{s_n})$ and $B_n(p_1,\ldots,p_n,r_1,\ldots,r_{t_n})$ such that the size of A_n and B_n is $n^{O(1)}$ and such that

$$U_n := U \cap \{0,1\}^n = \{(\delta_1,\ldots,\delta_n \in \{0,1\}^n | \exists \alpha_1,\ldots,\alpha_{s_n} A_n(\bar{\delta},\bar{\alpha}) \text{ holds}\}$$

and

$$V_n := V \cap \{0,1\}^n = \{(\delta_1,\ldots,\delta_n \in \{0,1\}^n | \exists \beta_1,\ldots,\beta_{t_n} A_n(\bar{\delta},\bar{\beta}) \text{ holds}\}.$$

Assuming that the sets U and V are disjoint sets is equivalent to the statement that the implications $A_n \to \neg B_n$ are all tautologies. Craig's interpolation theorem guarantees there is a formula $I_n(\bar{p})$ constructed only using atoms $\bar{p}$ such that

$$A_n \to I_n$$

and

$$I_n \to \neg B_n$$

are both tautologies. Thus the set

$$W := \bigcup_n \{\bar{\delta} \in \{0,1\}^n | I_n(\bar{\delta}) \text{ holds}\}$$

defined by the interpolant I_n separates U from V: $U \subseteq W$ and $W \cap V = \emptyset$. Hence an estimate of the complexity of propositional interpolation formulas in terms of the complexity of an implication yields an estimate to the computational complexity of a set separating U from V. In particular, a lower bound to a complexity of interpolating formulas gives also a lower bound on the complexity of sets separating disjoint $\mathcal{NP}$-sets. Of course, we cannot really expect to polynomially bound the size of a formula or a circuit defining a suitable W from the length of the implication $A_n \to \neg B_n$. This is because, as remarked by Mundici (45), it would imply that $\mathcal{NP} \cap co\mathcal{NP} \subseteq \mathcal{P}/poly$. In fact, for $U \in \mathcal{NP} \cap co\mathcal{NP}$ we can take V to be the complement of U and hence it must hold that $W = U$. In (39), Krajíček formulated the idea of effective interpolation as follows:

For a given propositional proof system, try to estimate the circuit-size of an interpolant of an implication in terms of the size of the shortest proof of the implication.

In other words, for a given propositional proof system establish an upper bound on the computational complexity of an interpolant of A and B in terms of the size of a proof of the validity of $A_n \to \neg B_n$. Then any pair A and B which is hard to interpolate yields a formula which must have large proofs of validity. This fact can be exploited in proving lower bounds, and indeed several new lower bounds came out from its application, see (41), (52). Besides lower bounds, effective interpolation revealad to be an excellent idea in other areas in proving results of independence in bounded arithmetic (53) and in establishing links between proof complexity and modern cryptography.The footnote 15 can stay.[15]

Definition 2.16. *A propositional proof system P admits effective interpolation if and only if there is a polynomial $p(x)$ such that any implication $A \to B$ with a proof in P of size m has an interpolant of a circuit size $\leq p(m)$.*

[15] A general overview of these applications has been given in (38) and (51).

The main point of the effective interpolation method is that by establishing a good upper bound for a proof system P in the form of the effective interpolation we prove lower bounds on the size of the proofs in P. That is,

Theorem 2.17. *Assume that U and V are two disjoints $\mathcal{NP}$-sets such that U_n and V_n are inseparable by a set of circuit complexity $\leq s(n)$, all $n \geq 1$. Assume that P admits effective interpolation. Then the implications $A_n \rightarrow \neg B_n$ require proofs in P of size $\geq s(n)^{\epsilon}$, for some $\epsilon > 0$.*

The system Resolution admits feasible interpolation, as it was proven in (41). In fact,

Theorem 2.18 (Krajíček (41))**.** *Assume that the set of clauses*

$$\{A_1, ..., A_m, B_1, ..., B_l\}$$

where

1. $A_i \subseteq \{p_1, \neg p_1, ..., p_n, \neg p_n, q_1, \neg q_1, ..., q_s, \neg q_s\}$, *all* $i \leq m$
2. $B_j \subseteq \{p_1, \neg p_1, ..., p_n, \neg p_n, r_1, \neg r_1, ..., r_t, \neg r_t\}$, *all* $j \leq l$

has a resolution refutation with k clauses.
Then the implication

$$\bigwedge_{i \leq m} (\bigvee A_i) \rightarrow \neg \bigwedge_{i \leq l} (\bigvee B_j)$$

has an interpolant I whose circuit-size is $kn^{O(1)}$.

The key idea of the proof of the previous theorem is that the structure of a resolution refutation allows one easily to decide which clauses cause the unsatisfiability under a specific assignment. Furthermore, it should be noticed that the interpolant can be computed by a polynomial time algorithm having an access to the resolution refutation. Theorem 2.18 is important because it is a theorem used later on in the next section.

2.4 "Mathematical" proof systems

The set of propositional tautologies $TAUT$ is a $co\mathcal{NP}$-complete set. In general a proof system is a relation $R(x, y)$ computable in polynomial time such that

$$x \in TAUT \text{ if and only if } \exists y(R(x, y)).$$

A proof of x is a y such that $R(x, y)$ holds. Thus one can take an $co\mathcal{NP}$-complete set and a suitable relation R over it and investigate the complexity of such proofs. In this section we recall a few proof systems (only one in some detail) "mathematically" based on $co\mathcal{NP}$-complete sets.
A nice example of a well-known "mathematical[16] " proof system is the proof system Cutting Plane CP. The Cutting Plane proof system (CP) is a refutation system based on showing the non-existence of solutions for a family of linear equalities. A line in a proof in th system CP is an expression of the form

$$\sum a_i \cdot x_i \geq B$$

where a_1,..., a_n, B are integers. Then for a given clause C, and the variables x_i, $i \in P$, occur positively in C, and variables x_i, $i \in N$, occur negatively in C, then C is represented by the linear inequality

$$\sum_{i \in P} x_i - \sum_{i \in N} x_i \geq 1 - |N|.$$

[16] This expression is taken from Pudlák (51).

A *CNF* formula is represented by the family of linear inequalities corresponding to its clauses. Thus for example the formula $(x_1 \vee \bar{x_2} \vee x_3) \wedge (\bar{x_1} \vee x_3 \vee x_4 \vee \bar{x_5})$ is represented by the inequalities $x_1 - x_2 + x_3 \geq 0$ and $-x_1 + x_3 + x_4 - x_5 \geq -1$. The axioms of the proof system are $x_i \geq 0$, $-x_i \geq -1$. The rules of inference are:

(a)
$$\frac{\sum a_i \cdot x_i \geq A \;\; \sum b_i \cdot x_i \geq B}{\sum (a_i + b_i) \cdot x_i \geq A + B}$$

(b)
$$\frac{\sum a_i \cdot x_i \geq A}{\sum (c \cdot a_i) \cdot x_i \geq c \cdot A}$$
where $c \geq 1$ is an arbitrary integer;

(c)
$$\frac{\sum (c \cdot a_i) \cdot x_i \geq A}{\sum a_i \cdot x_i \geq \left\lceil \frac{A}{c} \right\rceil}$$
where $c > 1$ is an arbitrary integer.

A derivation D of the inequality I from inequalities I_1, ..., I_m is a sequence D_1,...,D_n such that $I = D_n$ and for all $i < n$ either D_i is an axiom, or one of I_i, ..., I_m or inferred from D_j, D_k for $j, k < i$ by means of a rule of inference. A *CP* refutation of I_1,...,I_m is a derivation of $0 \geq 1$ from I_1, ..., I_m.
We have seen above the soundness and the completeness of Resolution for *CNF* formulas, see Theorem 2.15. Soundness in the sense that given any formula ϕ which has a resolution refutation π, ϕ is not satisfiable and completeness in the sense that given any unsatisfiable formula ϕ, there is a resolution refutation π of ϕ. Theorem 2.19 can be proved exploiting the completeness of *R*, since *CP* easily simulates resolution as observed in (23).

Theorem 2.19. *The proof system CP is sound and complete with respect to CNF formulas.*

For the soundness part we can argue as follows. Let ϕ be a *CNF* formula with a *CP* refutation γ. Suppose ϕ is satisfied by the assignment α. Instantiate each inequality in γ of ϕ by assigning the boolean variables their value under α. By induction on the length of γ we can prove that each instantiated inequality in the refutation γ holds. This is contradiction, because we cannot have the inequality $0 \geq 1$ as the last element of the refutation. Goerdt (30) proved that Frege systems polynomially simulate the *CP* proof system.
Other examples of mathematical proof systems are for instance the Nullstellensatz system introduced in (4), the Polynomial Calculus (15) and the Gaussian Calculus first defined in (5). At the end of this chapter a new mathematical proof system using cellular automata will be proposed.

3. Applications of propositional logic to cellular automata

Cellular automata can be described as large collections of simple objects locally interacting with each other. A *d*-dimensional cellular automaton consists of an infinite *d*-dimensional array of identical cells. Each cell is always in one state from a finite state set. The cells change their states synchronously in discrete time steps according to a local rule. The rule gives the

new state of each cell as a function of the old states of some finitely many nearby cells, its neighbours. In the literature one can find different types of neighbourods depening upon which group of cells are taken into consideration during the application of the local rule. In the figure below the Von Neumann, Moore and Simth neighbourods are displayed in Figure 1.

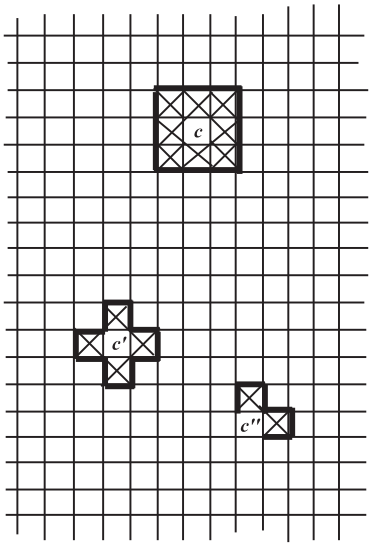


Fig. 1. The Moore neighborhood of the cell c, the von Neumann neighborhood of the cell c' and the Smith neighborhood of the cell c''.

The automaton is homogeneous so that all its cells operate under the same local rule. The states of the cells in the array are described by a configuration. A configuration can be considered as the state of the whole array. The local rule of the automaton induces a global function that tells how each configuration is changed in one time step. [17] In literature cellular automata take various names according to the way they are used. They can be employed as computation models (27) or models of natural phenomena (61), but also as tessellations structures, iterative circuits (12), or iterative arrays (18). The study of this computation model was initiated by von Neumann in the ′40s (46), (47). He introduced cellular automata as possible universal computing devices capable of mechanically reproducing themselves. Since that time cellular automata have also aquired some popularity as models for massively parallel computations.
Cellular automata have been extensively studied as discrete models for natural system they have several basic properties of the physical world: they are massively parallel, homogeneous and all interactions are local. Other physical properties such as reversibility and conservation laws can be programmed by selecting the local rule suitably. They provide very simple models of complex natural systems encountered in physics and biology. As natural systems they consists of large numbers of very simple basic components that together produce the complex behaviour of the system. Then, in some sense, it is not surprising that several physical systems

[17] For surveys on cellular automata the interested reader can see (35), (36).

(spin systems, crystal growth process, lattice gasses, ...) have been modelled using these devices, see (61).
In this section we show how the study of propositional proof complexity and some of its techniques can be exploited in order to investigate cellular automata and their properties. In this section we focus on a new proof of a foundamental theorem in the field, the Richardson theorem (54), given in (14). Then application of feasible interpolation shows how to find computational description of inverse cellular automata. Then we consider in some detail two complexity problems formulated in (28) and solved in (14) as well.

3.1 Cellular Automata: definitions and some basic results

From a formal point of view a cellular automaton is an infinite lattice of finite automata, called cells. The cells are located at the integer lattice points of the d-dimensional Euclidean space. In general any Abelian group $\mathcal{G}$ in place of $\mathbb{Z}^d$ can be used. In particular, we may consider $(\mathbb{Z}/m)^d$, a toroidal space, where $\mathbb{Z}/m$ is the additive group of integers modulo m. In $\mathbb{Z}^d$ we identify the cells by their coordinates. This means that the cells are adressed by the elements of $\mathbb{Z}^d$.

Definition 3.1. *Let S be a finite set of states and $S \neq \emptyset$. A configuration of the cellular automaton is a function $c : \mathbb{Z}^d \to S$. The set of all configurations is denoted by $\boldsymbol{C}$.*

At discrete time steps the cells change their states synchronously. Simply the next state of each cell depends on the current states of the neighboring cells according to an update rule. All the cells use the same rule, and the rule is applied to all cells in the same time. The neighboring cells may be the nearest cells surrounding the cell, but more general neighborhoods can be specified by giving the relative offsets of the neighbors.

Definition 3.2. *Let $N = (\vec{x_1}, ..., \vec{x_n})$ be a vector of n elements of $\mathbb{Z}^d$. Then the neighbors of a cell at location $\vec{x} \in \mathbb{Z}^d$ are the n cells at locations $\vec{x} + \vec{x_i}$, for $i = 1, ..., n$.*

The local transformation rule (transition function) is a function $f : S^n \to S$ where n is the size of the neighborhood. State $f(a_1, ..., a_n)$ is the new state of a cell at time $t + 1$ whose n neighbours were at states $a_1, ..., a_n$ at time t.

Definition 3.3. *A local transition function defines a global function $G : C \to C$ as follows,*

$$G(c)(\vec{x}) := f(c(\vec{x} + \vec{x_1}), \ldots, c(\vec{x} + \vec{x_n})).$$

The cellular automataton evolves from a starting configuration c^0 (at time 0), where the configuration c^{t+1} at time $(t + 1)$ is determined by c^t (at time t) by,

$$c^{t+1} := G(c^t).$$

Thus, cellular automata are dynamical systems that are updated locally and are homogeneous and discrete in time and space. Often in literature cellular automata are specified by a quadruple

$$\mathbb{A} = (d, S, N, f),$$

where d is a positive integer, S is the set of states (finite), $N \in (\mathbb{Z}^d)^n$ is the neighborhood vector, and $f : S^n \to S$ is the local transformation rule.

Definition 3.4. *A cellular automaton $\mathbb{A}$ is said to be injective if and only if its global function $G_{\mathbb{A}}$ is one-to-one. A cellular automaton $\mathbb{A}$ is said to be surjective if and only if its global function $G_{\mathbb{A}}$ is onto. A cellular automaton $\mathbb{A}$ is bijective if its global function $G_{\mathbb{A}}$ is one-to-one and onto.*

Let $\mathbb{A}$ and $\mathbb{B}$ be cellular automata. Let $G_{\mathbb{A}}$ and $G_{\mathbb{B}}$ the two global functions. Suppose that d is the same for $\mathbb{A}$ and $\mathbb{B}$ and that they have in common also S. We may compose $\mathbb{A}$ with $\mathbb{B}$ as follows: first run $\mathbb{A}$ and then run $\mathbb{B}$. Denoting the resulting cellular automaton by $\mathbb{B} \circ \mathbb{A}$ we have

$$G_{\mathbb{B}\circ\mathbb{A}} = G_{\mathbb{B}} \circ G_{\mathbb{A}}.$$

Notice that this composition can be formed effectively. If $N_{\mathbb{A}}$ and $N_{\mathbb{B}}$ are neighborhoods of $\mathbb{A}$ and $\mathbb{B}$, and $G_{\mathbb{A}}$ and $G_{\mathbb{B}}$ the global functions, then a neighborhood of $G_{\mathbb{B}} \circ G_{\mathbb{A}}$ consists of vectors $\vec{x} + \vec{y}$ for all $\vec{x} \in N_{\mathbb{A}}$ and $\vec{y} \in N_{\mathbb{B}}$.
The reader can see that the problem of establishing whether or not two given cellular automata $\mathbb{A}$ and $\mathbb{B}$, with $G_{\mathbb{A}}$ and $G_{\mathbb{B}}$, are equivalent is decidable. To see this it is enough to observe that:

(i) if $N_{\mathbb{A}} = N_{\mathbb{B}}$ then the local transformation rules, $f_{\mathbb{A}}$ and $f_{\mathbb{B}}$, are identical;

(ii) if $N_{\mathbb{A}} \neq N_{\mathbb{B}}$ then one can take $N_{\mathbb{A}} \cup N_{\mathbb{B}}$ and to test whether $\mathbb{A}$ and $\mathbb{B}$ agree on the expanded neighborhood.

Intuitively, the shift functions translate the configurations one cell down in one of the coordinate direction. Formally, for each dimension $i = 1, ..., d$ there is a corresponding shift function σ_i whose neighborhood contains only the unit coordinate vector $\vec{e_i}$ whose rule is the identity function id.[18] Translations are compositions of shift functions.
In the literature quite often a particular state $q \in S$ is specified as a quiescent state (which usually simulates empty cells). The state must be stable, i.e. $f(q, q, ..., q) = q$. Thus a configuration c is said to be quiescent if all its cells are quiescent, $c(\vec{x}) = q$.

Definition 3.5. *A configuration $c \in S^{\mathbb{Z}^d}$ is finite if only a finite number of cells are non-quiescent, i.e. the set (support),*

$$\{\vec{x} \in \mathbb{Z}^d | \; c(\vec{x}) \neq q\}$$

is finite.

Let $\mathbf{C}_F$ be the subset of $\mathbf{C}$ that contains only the finite configurations. Finite configurations remain finite in the evolution of the cellular automaton, because of the stability of q, hence the restriction G^F of G on the finite configurations is a function $G^F : \mathbf{C}_F \to \mathbf{C}_F$.

Definition 3.6. *A spatially periodic configuration is a configuration that is invariant under d linearly independent translations.*

This is equivalent to the existence of d positive integers $t_1, ..., t_d$ such that $c = \sigma_i^{t_i}(c)$ for every $i = 1, ..., d$. We denote the set of periodic configurations by $\mathbf{C}_P$. The restriction of G^P of G on the periodic configurations is hence a function $G^P : \mathbf{C}_P \to \mathbf{C}_P$.
Finite and periodic configurations are used in effective simulations of cellular automata on computers. Periodic configuarations are referred to as the periodic boundary conditions on a finite cellular array. For instance, when $d = 2$, this is equivalent to running the cellular automaton on a torus (see Figure 2) that is obtained by joining together the opposite sides of a rectangle. The relevant group is $(\mathbb{Z}/t_1) \times (\mathbb{Z}/t_2)$.

Definition 3.7. *Let $\mathbb{A}$ be a cellular automaton. A configuration c is called a Garden of Eden configuration of $\mathbb{A}$, if c is not in the range of the global function $G_{\mathbb{A}}$.*

[18] The one-dimensional shift function is the left shift $\sigma = \sigma_1$

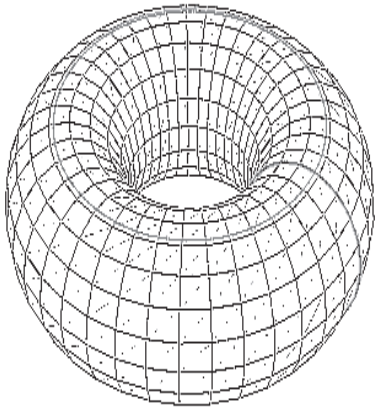


Fig. 2. A toroidal arrangement: when one goes off the top, one comes in at the corresponding position on the bottom, and when one goes off the left, one comes in on the right.

When one deals with finate sets a basic property that holds is the following: a function from a finite set into itself is injective if and only if the function is surjective. This property is partially tru for cellular automata. In fact, an injective cellular automaton is always surjective, but the converse does not hold. When we consider finite configurations the behaviour is more analogous to finite sets. Theorem 3.9, a combination, a combination of two results proved by Moore (43) and by Myhill (44) respectively, shows out exaclty this fact.

Definition 3.8. *A pattern α is a function $\alpha : P \to S$, where $P \subseteq \mathbb{Z}^d$ is a finite set. Pattern α agrees with a configuration c if and only if $c(\overline{x}) = \alpha(\overline{x})$ for all $\overline{x} \in P$.*

Theorem 3.9 (Moore (43), Myhill (44))**.** *Let $\mathbb{A}$ be a cellular automaton. Then $G^F_{\mathbb{A}}$ is injective if and only if $G^F_{\mathbb{A}}$ has the property that for any given pattern α there exists a configuration c in the range of $G^F_{\mathbb{A}}$ such that α agrees with c.*

The proof of the theorem is combinatorial and holds for any dimension d. The following theorem summarizes the situation regarding the injectivity and the surjectivity of cellular automata.

Theorem 3.10 (Richardson (54))**.** *Let $\mathbb{A}$ be a cellular a automaton. Let $G_{\mathbb{A}}$ be its global function and $G^F_{\mathbb{A}}$ be $G_{\mathbb{A}}$ restricted to the finite configurations. Then the following implications hold:*

1. *If $G_{\mathbb{A}}$ is one-to-one then $G^F_{\mathbb{A}}$ is onto.*
2. *If $G^F_{\mathbb{A}}$ is onto then $G^F_{\mathbb{A}}$ is one-to-one.*
3. *$G^F_{\mathbb{A}}$ is one-to-one if and only if $G_{\mathbb{A}}$ is onto.*

Definition 3.11. *A cellular automaton $\mathbb{A}$ with global function $G_{\mathbb{A}}$ is invertible if there exists a cellular automaton $\mathbb{B}$ with global function $G_{\mathbb{B}}$, such that $G_{\mathbb{B}} \circ G_{\mathbb{A}} = id$, where id is the identity function on $\boldsymbol{C}$.*

It is decidable whether two given cellular automata $\mathbb{A}$ and $\mathbb{B}$ are inverses of each other. This is a consequence of the effectiveness of the composition and the decidability of the equivalence. In 1972 Richardson proved the following important theorem about cellular automata:[19]

Theorem 3.12 (Richardson (54))**.** *Let $\mathbb{A}$ be an injective cellular automaton. Then $\mathbb{A}$ is bijective and the inverse of $G_{\mathbb{A}}$, $G_{\mathbb{A}}^{-1}$, is the global function of a cellular automaton.*

The same year of Richardson's result Amoroso and Patt proved the following theorem:

Theorem 3.13 (Amoroso and Patt (1))**.** *Let $d = 1$. Then there exists an algorithm that determines, given a cellular automaton $\mathbb{A} = (1, S, N, f)$, if $\mathbb{A}$ is invertible or not.*

In the same paper they also provided an algorithm in order to determine if a given cellular automaton is surjective.[20] In higher spaces the problem of showing if a given cellular automaton is surjective or not, has been shown undecidable by Kari (34). Kari proved also that the reversibility of cellular automata is undecidable too,

Theorem 3.14 (Kari (34))**.** *Let $d > 1$. Then there is no an algorithm that determines, given $\mathbb{A} = (d > 1, S, N, f)$, if $\mathbb{A}$ is invertible or not.*

The proof of Theorem 3.14 is based on the transformation of the tiling problem shown to be undecidable by Berger (8), into the invertibility problem on a suitable class of cellular automata.

3.2 A new proof of the Richardson theorem based on propositional logic

Theorem 3.12 was proven using a topological argument in combination with the Garden of Eden theorem. Richardson's proof was non-constructive (it used compactness of a certain topological space) and the new proof is formally non-constructive too (becaue of the use of compactness of propositional logic). One should notice that this non-constructivity is unavoidable because of the undecidibility theorem proved by Kari; see Theorem 3.14. The new proof given in (14) offers a technical simplification: only basic logic is involved requiring straightforward formalism, and it allows us to apply an interpolation theorem. It is important to point out that the proof can be made fully constructive, if we consider periodic configurations; considering $d = 2$ the working space becomes a torus.[21]
The proof use as a dimension $d = 2$ and on the binary alphabet. This simplifies the notation but displays the idea of the proof in full generality which works $d > 2$ and extended alphabets as well. We report below the statement of Richardson's theorem and the proof in some detail. The reader can see (14) for the complete one.

Theorem 3.15. *Let $\mathbb{A}$ be a cellular automata over $\mathbb{Z}^2$ (with 0, 1 alphabet) whose global function $G_{\mathbb{A}}$ is injective. Then there is a cellular automata $\mathbb{B}$ (with 0, 1 alphabet) with global function $G_{\mathbb{B}}$ such that $G_{\mathbb{B}} \circ G_{\mathbb{A}} = id$.*

The new proof given by the present author in (14) goes as follows. For an n-tuple of $\mathbb{Z}^2$-points $N = ((u_1, v_1), \ldots, (u_n, v_n))$ defining the neighborhood of $\mathbb{A}$ denoted by

$$(i, j) + N$$

[19] Recall that on finite configurations the global function may be onto and one-to-one even if the cellular automaton is not reversible.
[20] Later Sutner designed elegant decision algorithms based on de Bruijn graphs, see (60).
[21] After Theorem 3.15 this point will be discussed in some detail.

the n-tuple $(i+u_1, j+v_1), \ldots, (i+u_n, j+v_n)$. Then we define a suitable embedding into propositional logic as follows: For each i, j let $p_{i,j}$ be a propositional variable. Denote by $p_{(i,j)+N}$ the n-tuple of variables

$$p_{i+u_1,j+v_1}, \ldots, p_{i+u_n,j+v_n}.$$

This embedding has the consequence of characterizing the transformation function of the cellular automata a as a boolean function of n-variables

$$p^{t+1}_{(i,j)} = f(p^t_{(i,j)+N})$$

where the superscript t and $t+1$ denote the discrete time. An array

$$(r_{(i,j)})_{(i,j)\in\mathbb{Z}^2}$$

(we shall skip the indices and write simply $\vec{r}$) of 0 and 1 describes the configurations obtained by $G_{\mathbb{A}}$ from an array

$$(p_{(i,j)})_{(i,j)\in\mathbb{Z}^2}$$

if and only if conditions

$$r_{(i,j)} = f(p_{(i,j)+N}),$$

for all (i, j) are satisfied.
Denote the infinite set of all these conditions $T_A(\vec{p}, \vec{r})$.
Then the next step is to define f by a CNF (or DNF) formula. In this manner we can think of $T_A(\vec{p}, \vec{r})$ as of a propositional theory consisting of clauses.
A basic observation is that the injectivity of $G_{\mathbb{A}}$ is equivalent to the fact that the theory

$$T(\vec{p}, \vec{r}) \cup T(\vec{q}, \vec{r})$$

(where $\vec{p}$, $\vec{q}$ and $\vec{r}$ are disjoints arrays of variables) logically implies all equivalences of the form:

$$p_{(i,j)} \equiv q_{(i,j)}$$

for all $(i, j) \in \mathbb{Z}^2$. Since the theory is unaltered if we replace all indices (i, j), by $(i, j) + (i_0, j_0)$, (any fixed $(i_0, j_0) \in \mathbb{Z}^2$) this is equivalent to the fact that

$$T(\vec{p}, \vec{r}) \cup T(\vec{q}, \vec{r})$$

implies that

$$p_{(0,0)} \equiv q_{(0,0)}.$$

This can be reformulated in the following manner:

$$T(\vec{p}, \vec{r}) \cup \{p_{(0,0)}\} \cup T(\vec{q}, \vec{r}) \cup \{\neg q_{(0,0)}\}$$

is unsatisfiable.
By applying the compactness theorem for propositional logic to deduce that there are finite theories

$$T_0(\vec{p}, \vec{r}) \subseteq T(\vec{p}, \vec{r})$$

and

$$T_0(\vec{q}, \vec{r}) \subseteq T(\vec{q}, \vec{r})$$

such that

$$T_0(\vec{p},\vec{r}) \cup \{p_{(0,0)}\} \cup T_0(\vec{q},\vec{r}) \cup \{\neg q_{(0,0)}\}$$

is unsatisfiable. At this point in the proof we can use the Craig interpolation theorem. This result guarantees the existence of a formula $I(\vec{r})$, such that:

$$T_0(\vec{p},\vec{r}) + p_{(0,0)} \vdash I(\vec{r})$$

and

$$I(\vec{r}) \vdash T_0(\vec{q},\vec{r}) \to q_{(0,0)}.$$

Although we write the whole array $\vec{r}$ in $I(\vec{r})$, the formula obviously contains only finitely many r variables.
Thus by application of the deuction theorem we have that:

$$T_0(\vec{p},\vec{r}) \vdash p_{(0,0)} \to I(\vec{r})$$

and

$$T_0(\vec{q},\vec{r}) \vdash I(\vec{r}) \to q_{(0,0)}.$$

Then after suitably renaming the propositional variables we see that the the second implication gives:

$$T_0(\vec{p},\vec{r}) \vdash I(\vec{r}) \to p_{(0,0)}$$

i.e. together

$$T_0(\vec{p},\vec{r}) \vdash I(\vec{r}) \equiv p_{(0,0)}.$$

The interpolant $I(\vec{r})$ computes the symbol of cell (0,0) in the configuration prior to $\vec{r}$. This it defines the inverse to $\mathbb{A}$. Let $M \subseteq \mathbb{Z}^2$ be the finite set of $(s,t) \in \mathbb{Z}^2$ such that $r_{(s,t)}$ appears in $I(\vec{r})$. Finally the inverse cellular automaton $\mathbb{B}$ as follows:

1. Alphabet is 0,1;
2. The neighborhood is M;
3. The transition function is given by $I(\vec{r})$,

This concludes the proof.
We conclude this part about the Richardson theorem with a few remarks about the proof sketched above.

1. The construction of the inverse cellular automaton carried out in the proof given in (14) has two key moments. First the application of the compactness theorem after a suitable embedding into propositional logic. In a second moment the interpolation theorem is applied leading to some kind of effectiveness. Of course, the application of compactness leads to a non-recursive procedure but as we previously noticed this fact is unavoidable because of theorem 3.14.
2. The construction guarantees that

   $$G_{\mathbb{B}}(G_{\mathbb{A}}(\vec{p})) = \vec{p}$$

   but it does not - a priori - imply that also

   $$G_{\mathbb{A}}(G_{\mathbb{B}}(\vec{r})) = \vec{r}.$$

This is a consequence of Theorem 3.9 (Garden of Eden Theorem).

3. If the interpolant $I(\vec{r})$ contains M variables (i.e. the neighborood of $\mathbb{B}$ has size $|M|$) then the size of $\mathbb{B}$ (as defined in (28), see Definition 3.16 below) is $O(2^{|M|})$. This also bounds the size $|I|$ of any formulas defining the interpolant, but the interpolant I could be in principle defined by a substantially smaller formula (e.g. of size $O(|M|)$.)

It is interesting to notice that the same argument works for the version of the previous theorem with $(\mathbb{Z}/m)^2$ in place of $\mathbb{Z}^2$. In this case, the starting theory $T(\vec{p},\vec{r})$ is finite: of size $O(m^2\, 2^n)$ where m^2 is the size of $(\mathbb{Z}/m)^2$ and $O(2^n)$ bound the sizes of CNFs/DNFs formulas for the transition function of $\mathbb{A}$. In this case we can avoid the use of the compactness theorem and the interpolation can be directly applied.
In (14) is also given a constructive way how to to find the interpolant and the inverse automaton as well.

3.3 Some new complexity results

In this paper Durand (28) proved the first complexity results concerning a global property of cellular automata of dimension $\geq$ 2 (see Theorem 3.17). By Kari's result (33) the reversibility of a cellular automaton with $d \geq 2$ is not decidable. This implies that the inverse of a given cellular automaton cannot be found by an algorithm: its size can be greater than any computable function of the size of the reversible cellular automaton. Durand's result shows that even if we restrain the field of action of cellular automata (with $d = 2$) to finite configuration bounded in size, it is still very hard to prove that the cellular automaton is invertible or not: the set of cellular automata invertible on finite configurations is *co*$\mathcal{NP}$-complete (see below). Nevertheless ome open problems are left from Durand' work in (28). A solution to two of the open problems stated in (28) has been offered in (14) by the present author. Below we give in some detail a summary of Durand's theorem and of the two solution. For more details the reader can see (28) and (14). In (28) it is assumed that the size of a cellular automaton corresponds to the size of the table of its local function and of the size of its neighborhood. More precisely:

Definition 3.16. *If s is the number of states of a cellular automaton $\mathbb{A}$ and $N = (x_1, ..., x_n)$ then the size of a string necessary to code the table of the local function plus the vector N of $\mathbb{A}$ is $s^n \cdot \log(s) + o(s^n \cdot \log(s))$.*

Durand proved that the decision problem concerning invertibility of cellular automata of dimension 2 belongs to the class of *co*$\mathcal{NP}$-hard problems or to the class of *co*$\mathcal{NP}$-complete problems if some bound is introduced on the size of the finite configurations considered.[22]
For the *co*$\mathcal{NP}$-completeness we assume that the size of the neighborhood is lower than the size of the transition table of the cellular automaton, i.e. $\forall x \in N, |x| \leq s^n$.
Now, consider the following problem:

PROBLEM *(CA-FINITE-INJECTIVE):*
Instance: A 2-dimensional cellular automaton $\mathbb{A}$ with von Neumann neighborhood. Two integers p and q less than the size of $\mathbb{A}$.
Question: Is $\mathbb{A}$ injective when restricted to all finite configurations $\leq p \times q$?
The theorem below is the main result in (28),

[22] Notice that result is obtained for a 2-dimensional cellular automata with von Neumann neighborhood, see Figure 1.

Theorem 3.17 (Durand (28))**.** *The problem CA-FINITE-INJECTIVE is co$\mathcal{NP}$-complete.*

The proof is based on tiling. A tile is a square and its sides are colored. The colors belong to a finite set called the color set. All tiles have the same size. A plane tiling is valid if and only if all pairs of adjacent sides have the same color.[23] A finite tiling can be defined as follows. We assume that the set of colors contains a special "blank color" and that the set of tiles contains a ''blank tile" (a tile whose sides are blank.) A finite tiling is an almost everywhere blank tiling of the plane. If there exist two integers i and j such that all the nonblank tiles of the tiling are located inside a square of size $i \times j$, then we say that the size of the finite tiling is lower than $i \times j$. Inside the $i \times j$ square, there can be blank and nonblank tiles. If we have at least one nonblank tile, then the tiling is called nontrivial.
Durand in its proof introduces a special tile set δ. The sides contain a color ("blank", "border", "odd", "even", or "the-end"), a label (N, S, E, W, $N+$, $S+$, $E+$ $W+$, or ω), and possibly an arrow. A tiling is valid with respect to δ if and only if all pairs of adjacent sides have the same color, the same label, and for each arrow of the plane, its head points out the tail of an arrow in the adjacent cell. A basic rectangle of size $p \times q$ is a finite valid tiling of the plane of size $p \times q$ with no size labeled "blank" or "border" inside the rectangle.
Then, given a finite set of colors B with a blank color and a collection $\tau \in B^4$ of tiles including a blank tile, Durand constructs a cellular automaton $\mathbb{A}_\tau$ and proves the following basic theorem which provides a link between tilings and cellular automata:

Theorem 3.18 (Durand (28))**.** *Let $n \geq 3$ be an integer and τ be a set of tiles. The cellular automaton $\mathbb{A}_\tau$ is not injective restricted to finite configurations of size smaller than $2n \times 2n$ if and only if the tile set τ can be used to form a finite nontrivial tiling of the plane of size smaller than $(2n-4) \times (2n-4)$.*

Then using Theorem 3.18 he proves that ***PROBLEM*** *(CA-FINITE-INJECTIVE)* is *co$\mathcal{NP}$*-complete, i.e. Theorem 3.17.
Notice that if one drops the restriction on the bound of the size of the neighborhood then a proof of the *co$\mathcal{NP}$*-hardness of *CA-INFINITE-INJECTIVE* can be obtained; for more details on this the reader can see (28).
What is assumed in the previous result is basically that the size of the representation of a cellular automaton corresponds to the size of its transition table. Durand (28) asked if the *co$\mathcal{NP}$*-completeness result can be true also if we define the size of a cellular automaton as the length of the smallest program (circuit) which computes its transition table. A second question formulated in (28) is the following: suppose that we have an invertible cellular automaton given by a simple algorithm and that we restrict ourself to finite bounded configurations. Then is the inverse given by a simple algorithm too? In (14) both problem got a solution. Let us see in some detail the solutions. The second problem is solved for succintness on $d = 1$ it is fairly simple to get obtain examples for $\mathbb{Z}^2$ or $(\mathbb{Z}/m)^2$.
Let $f : \{0,1\}^n \to \{0,1\}^n$ be a Boolean function having the following properties:

1. f is a permutation;
2. f is computed by a polynomial size circuit.
3. The inverse function f^{-1} requires an exponential size circuit, $\exp(\Omega(n))$.

As an example of these type of functions one could take one-way permutations (e.g. conjecturally based on factoring or discrete logarithm). Now define a cellular automaton $\mathbb{A}_f$ as follows:

[23] Notice that is not allowed to turn tiles.

1. Alphabet: 0, 1, #.
2. Neighborhood of $i \in \mathbb{Z}$:

$$N = \langle i-n, i-n+1, \ldots, i, i+1, \ldots, i+n \rangle$$

i.e. $|N| = 2n+1$.

3. Transition function:
(i) $p_i^t = \# \to p_i^{t+1} = \#$
(ii) If $p_i^t \in \{0,1\}$ and there are j, k such that:
(a) $j < i < k$ and $k - j = n + 1$;
(b) $p_j^t = p_k^t = \#$
(c) $p_r^t \in \{0,1\}$ for $r = j+1, j+2, \ldots, i, \ldots, k-1$
define

$$p_i^{t+1} = (f(p_{j+1}^t, \ldots, p_{k-1}^t))_i$$

where $(f(p_{j+1}^t, \ldots, p_{k-1}^t))_i$ is the i-th bit of $f(p_{j+1}^t, \ldots, p_{k-1}^t)$.
(iii) If $p_i^t \in \{0,1\}$ and there are no j, k satisfying (ii) then put

$$p_i^{t+1} = p_i^t.$$

Informally the automaton $\mathbb{A}_f$ can be summarized as follows: every $0-1$ segment between two consecutives #'s that does not have the length exactly n is left unchanged. Segments of length n are trasformed according to the permutation f.
The inverse automaton $\mathbb{B}$ is defined in the same manner using f^{-1} in place of f ($\mathbb{B} = \mathbb{A}_{f^{-1}}$).

Theorem 3.19. *Assume that $f : 2^n \to 2^n$ is a permutation computable by a size poly(n) circuit such that any circuit computing the inverse function f^{-1} must have size at least $\exp(n^{\Omega(1)})$. Then the cellular automaton $\mathbb{A}_f$ is invertible but has an exponentially smaller circuit-size than its inverse cellular automaton.*

The proof is based on the fact that by the construction the inverse cellular automaton has a transition table which defines a boolean function which has a circuit-size exponential in n.
The hypothesis of Theorem 3.19 follows from the existence of cryptographic one-way functions. In particular, it follows from the exponential hardness of factoring or of discrete logarithm.
Thus Theorem 3.19 solves negatively one of the open problem formulated by Durand (28) that we have described above: a very "simple" algorithm giving a reversible cellular automaton (even if restricted to finite configurations) can have an inverse which is given by an algorithm which is exponentially bigger and then not "simple".[24]
The second problem stated in (28) and solved in (14) asks asks about *co$\mathcal{NP}$*-completeness of the injectivity of cellular automata when it is represented by a program (circuit) rather than by a transition table.
Consider the following problem:

PROBLEM (P1):
Input: A circuit $C(x_1, ..., x_n)$ defining the transition table function of $0-1$ cellular automaton

[24] Where "simple" algorithm means polynomial time algorithm.

$\mathbb{A}_C$ with a neighborhood N of size $|N| = n$.
*Question:*Is $\mathbb{A}_C$ injective on $\mathbb{Z}^1$?

Theorem 3.20. *Problem (P1) is co$\mathcal{NP}$-hard.*

The proof offered in (14) describes a polynomial reduction from $TAUT$ to *(P1)*. Let $\phi(x_1, ..., x_n)$ be a propositional formula. A sketch of the proof can be the following. Let the alphabet be 0,1 and the neighborhood N be $\langle 0, ..., n\rangle$. Now define the cellular automaton $\mathbb{A}$ as follows:

$$p_i^{t+1} := \begin{cases} p_i^t, & \text{if} \quad \phi(p_{i+1}^t, ..., p_{i+n}^t) \\ 0, & \text{otherwise.} \end{cases}$$

Clearly the circuit defining $\mathbb{A}$ is

$$p_i^t \wedge \phi(p_{i+1}^t, ..., p_{i+n}^t)$$

and has size $O(|\phi|)$. This means that the map $\phi \to \mathbb{A}$ is polynomial time.
If $\phi \in TAUT$ then always $p_i^{t+1} = p_i^t$. In this case $\mathbb{A}$ is a cellular automaton doing nothing, i.e. its global map is the identity and, in particular, it is invertible. Assume $\phi \notin TAUT$. Then two different configurations mapped by $\mathbb{A}$ to the same configuration have been constructed. Let $i_0 \geq 1$ be minimal i_0 such that there is a truth assignment $\bar{a} = (a_1, ..., a_n) \in \{0,1\}^n$ satisfying:
(i) $\neg\phi(\bar{a})$;
(ii) $a_{i_0} = \ldots = a_n = 0$;
(iii) either $i_0 = 1$ or $a_{i_0-1} = 1$.
Informally, $\bar{a}$ has the longest segment of 0's on the right hand side that is possible for assignments falsifying ϕ. Define two configurations as follows:

$$C_0 : \langle ..., 0, 0, 0, a_1, ..., a_n, 0, 0, ...\rangle$$

and

$$C_1 : \langle ..., 0, 0, 1, a_1, ..., a_n, 0, 0, ...\rangle.$$

The two configurations differ only in the position 0. Easily the theorem follows from the following lemma.

Lemma 3.21. *The two configurations C_0 and C_1 are both mapped by $\mathbb{A}$ to C_0.*

The proof is given using the definition above and then Theorem 3.20 follows directly from the application of the lemma.
Now, consider a finite modification of the problem *(P1)*:

PROBLEM (P2):
Input:

1. $\mathbb{A}_C$ as in *(P1)*;
2. $1^{(m)}$, such that $m > n$. (Notice that this condition implies that $\mathbb{A}_C$ is well-defined on $(\mathbb{Z}/m)$ tori.)

Question: Is $\mathbb{A}_C$ injective on $(\mathbb{Z}/m)$?

Theorem 3.22. *(P2) is co$\mathcal{NP}$-complete.*

The proof of Theorem 3.22 is quite similar to the proof of 3.20

4. Inverse Cellular Automata as propositional proofs

In this last section of the chapter we combine the Richardson theorem with the $co\mathcal{NP}$-completeness result of Durand (28) and we define a new type of a proof system $\mathbb{P}_{\mathbb{CA}}$. This proof system $\mathbb{P}_{\mathbb{CA}}$ is a proof system for the membership in a $co\mathcal{NP}$-complete language $\mathcal{L}_D$ (to be specified below). As the set $TAUT$ of propositional tautologies can be polynomially reduced to $\mathcal{L}_D$, $\mathbb{P}_{\mathbb{CA}}$ can be thought of also as a propositional proof system in the sense on Cook and Reckhow (21).
In this conclusive section we show that: there is polynomial time algorithm having a cellular automaton $\mathbb{A}$ with von Neumann neighborhood and a cellular automaton $\mathbb{B}$ with an arbitrary neighborhood and with the same alphabet of $\mathbb{A}$ as inputs, it can decide whether or not $\mathbb{B}$ is an inverse to $\mathbb{A}$. Then, we define a "mathematical" proof system for $\mathcal{L}_D$ satisfiyng the Cook and Reckhow definition (21). We conclude this chapter with some concluding remarks and some open questions when we consider our new proof system with respect to polynomial simulations.
We considered Durand's result of $co\mathcal{NP}$-completeness in the section 3. Let us recall the problem because it will be useful below. The problem that has been called (CA-FINITE-INJECTIVE) goes as follows:

PROBLEM *(CA-FINITE-INJECTIVE):*
Instance: A 2-dimensional cellular automaton $\mathbb{A}$ with von Neumann neighborhood. Two integers p and q smaller than the size of $\mathbb{A}$.
Question: Is $\mathbb{A}$ injective when restricted to all finite configurations $\leq p \times q$?
We shall also use the name CA-FINITE-INJECTIVE for the language of inputs with an affirmative answer.
Now we reformulate Durand's problem a bit in that we consider cellular automata operating on $(\mathbb{Z}/m)^2$ rather than on finite rectangles in $\mathbb{Z}^2$. We are replacing rectangles in $\mathbb{Z}^2$ by $(\mathbb{Z}/m)^2$ in order to be compatible with our treatement of Richardson's theorem given in the third chapter.
Consider a variant of the problem *CA-FINITE-INJECTIVE* in which the cellular automata operate on $(\mathbb{Z}/m)^2$ rather than on "finite configurations". We call this problem ***PROBLEM****(CA-TORI-INJECTIVE):*

PROBLEM*(CA-TORI-INJECTIVE):*
Instance: A 2-dimensional cellular automaton $\mathbb{A}$ with von Neumann neighborhood and $m \geq 3$, m is smaller than the size of $\mathbb{A}$.
Question: Is $\mathbb{A}$ injective when restricted to $(\mathbb{Z}/m)^2$?

Definition 4.1. *The language $\mathcal{L}_D$ is the set of pairs $(m, \mathbb{A})$ for which the* **PROBLEM**(CA-TORI-INJECTIVE) *has an affirmative answer.*

In terms of languages the problem above will be called $\mathcal{L}_D$. Thus, of course Theorem 3.17 by Durand can be simply stated as follows:

Theorem 4.2. *$\mathcal{L}_D$ is a $co\mathcal{NP}$-complete language.*

Now, consider the following lemma which states the existence of a polynomial time algorithm deciding the inverse:

Lemma 4.3. *There is a polynomial time algorithm that on the two inputs:*

1. *a cellular automaton* $\mathbb{A}$ *with von Neumann neighborhood;*
2. *a cellular automaton* $\mathbb{B}$ *with an arbitrary neighborhood and the same alphabet as* $\mathbb{A}$,

decides whether or not $\mathbb{B}$ *is an inverse to* $\mathbb{A}$.

Proof. The automata $\mathbb{A}$ and $\mathbb{B}$ are presented to the algorithm by the tables of their local functions, see Definition 3.16. Assume that the alphabet of $\mathbb{A}$ and $\mathbb{B}$ has S symbols and that the size of $\mathbb{B}$ neighborhood is N. Hence the size of $\mathbb{A}$ and $\mathbb{B}$ are $O(S^5 \cdot \log(S))$ and $O(S^N \cdot \log(S))$, respectively.
To evaluate a cell (i, j) in $\mathbb{B} \circ \mathbb{A}$ we need to look at a von Neumann neighborhood of all N points in the neighborhood of (i, j) in $\mathbb{B}$, i. e. on at most $\leq 5N$ cells. Considering all the possible $\leq S^{5N}$ patterns on these cells yields in a list of all possible patterns ($\leq S^N$) on the neighborhood of (i, j) in $\mathbb{B}$, after the action of $\mathbb{A}$. Then we check that in all these patterns $\mathbb{B}$ produces in the cell (i, j) the original symbol.
The time they need is bounded above by $O(S^{5N} \cdot (N \cdot S^5 \cdot \log(S)) \cdot (S^N \log(S)) = S^{O(N)})$, where S^{5N} bounds the number of patterns to check, $N \cdot S^5 \cdot \log(S)$ bounds the time need to compute the pattern on the neighborhood of (i, j) in $\mathbb{B}$ after the action of $\mathbb{A}$ (for any fixed pattern), and $S^N \cdot \log(S)$ bounds the time need to compute the symbol of (i, j) after the action of $\mathbb{B}$. However, the quantity $S^{O(N)}$ is polynomial in terms of the size of $\mathbb{B}$, i.e. the algorithm is polynomial time.

□

Let us remark that the restriction on $\mathbb{A}$ to a von Neumann neighborhood is essential. If $\mathbb{A}$ was allowed to have an arbitrarily neighborhood M, then the algorithm would need time $S^{O(M \cdot N)}$ which is only quasi-polynomial in the sizes $O(S^M \cdot \log(S))$ and $O(S^N \cdot \log(S))$ of the inputs $\mathbb{A}$ and $\mathbb{B}$.

4.1 A proof system based on cellular automata

In this section we define a new proof system $\mathbb{P}_{\mathbf{CA}}$ based on cellular automata. As far as we know this is the first proof system based on cellular automata.

Definition 4.4. $\mathbb{P}_{\mathbf{CA}}$ *is a proof system for the language* $\mathcal{L}_D$. *A* $\mathbb{P}_{\mathbf{CA}}$ *proof for the pair* $(m, \mathbb{A}) \in \mathcal{L}_D$ *is cellular automaton* $\mathbb{B}$ *such that:*

1. $\mathbb{B}$ *has the same alphabet as* $\mathbb{A}$*;*
2. $\mathbb{B}$ *is inverse to* $\mathbb{A}$.

Lemma 4.5. $\mathbb{P}_{\mathbf{CA}}$ *is a proof system for the language* $\mathcal{L}_D$.

Proof. If $\mathbb{A} \in \mathcal{L}_D$, then a suitable cellular automaton $\mathbb{B}$ exists by Richardson's theorem, Theorem 3.15. On the other hand the existence of $\mathbb{B}$ implies that the cellular automaton $\mathbb{A}$ is injective, i.e. $\mathbb{A} \in \mathcal{L}_D$. Hence $\mathbb{P}_{\mathbf{CA}}$ is complete and sound.
Finally, the provability relation is polynomial time decidable by Lemma 4.3.

□

The statement $\mathbb{A} \in \mathcal{L}_D$ can be expressed in a propositional way, same as in the proof of Richardson's theorem proved in (14). In particular, a proof of $\mathbb{A} \in \mathcal{L}_D$ is a proof of the unsatisfiability of the formula[25]:

$$T_0(\vec{p}, \vec{r}) \cup \{p_{(0,0)}\} \cup T_0(\vec{q}, \vec{r}) \cup \{\neg q_{(0,0)}\}$$

[25] See top page 66, the formula denoted by (*).

Hence any propositional proof system Q can be thought of also as a proof system for $\mathcal{L}_D$: a proof is a proof in Q of this formula.
We may observe at this place that having in particular a resolution proof of the formula gives us at least a circuit that describes the transition function of the inverse cellular automaton and whose size is polynomial in the size of the resolution proof: feasible interpolation allows to extract a circuit computing the interpolant and the interpolant defines the transition function. We remark that this leads to an interesting question about feasible interpolation. The size of the inverse automaton $\mathbb{B}$ is $O(S^N \log(S))$ where S is the size of the alphabet (common with the cellular automaton $\mathbb{A}$) and N is the size of the neighborhood of the inverse cellular automaton $\mathbb{B}$. Hence it is the quantity N that we would like to estimate. For this it would be very useful to have an estimate on the number of atoms the interpolant (produced by the feasible interpolation method or by any other specific method) depends on.

4.2 Some concluding remarks regarding the new proof system

The main problems which remain open from this last part are the followings: can we establish some polynomial simulation between $\mathbb{P}_{\mathbb{CA}}$ and some existing proof system such as Resolution? The investigation of this problem is hampered by the convoluted proof of Durand's theorem; a good place where to start thus would be to find a simple (or at least a simpler) proof of the latter. It would be desiderable to have a proof which involves propositional logic, as the proof of Richardson's theorem given in (14), since this could give us a very elegant and unified framework.
Having such a simplified proof one could use the well-known relation between bounded arithmetic and proof systems (see (37)) and attempt to prove the soundness of $\mathbb{P}_{\mathbb{CA}}$ in the theory corresponding to R. Such a soundness proof would imply polynomial simulation of $\mathbb{P}_{\mathbb{CA}}$ by R via a universal argument. We remark that the proof of Durand's theorem appears to be formalizable in the theory V^0, if that is indeed the case this would imply a polynomial simulation of $\mathbb{P}_{\mathbb{CA}}$ by a constant-depth Frege system. [26]

5. Conclusions

In this chapter we have seen some interactions between the study of cellular automata and the study of propositional proofs. In some sense the first attempt was made in (14) and the ending part of the final section constitutes a development in that direction. Nevertheless as pointed out at the end of the previous section several problems concerning simulations remain open for the reasons given above.

6. Acknowledgements

I thank Jan Krajíček for helpful discussions.

7. References

[1] S. Amoroso and Y.N. Patt, Decision procedures for surjectivity and injectivity of parallel maps for tasselation structures, *Jour. Comput. System Scie.*, 6, (1972), 448-464.
[2] E. Berlekamp, J. Conway, R. Elwyn and R. Guy, *Winning way for your mathematical plays*, vol. 2, Academic Press, (1982).

[26] See (37) for an extensive background on bounded arithmetic.

[3] P. Beame, H. Kautz and A. Sabharwal, Towards Understanding and Harnessing the Potential of Clause Learning, *Journal of Artificial Intelligence Research* (JAIR), 22, (2004), pp. 319-351.
[4] P. Beame, R. Impagliazzo, J. Krajíček, T. Pitassi, and P. Pudlák, Lower bounds on Hilbert's Nullstellensatz and propositional proofs, in *Proc. London Math. Soc.*, 73(3), (1996), pp. 1-26.
[5] E. Ben-Sasson and R. Impagliazzo, Random CNF'S are hard for the polynomial calculus, in *Proceedings of the 40th Annual IEEE Symposium on Foundations of Computer Science*, (1999).
[6] E. Ben-Sasson, R. Impagliazzo, and A. Wigderson, Near-optimal separation of tree-like and general resolution, *ECCC*, Report TR02-005, (2000).
[7] E. Ben-Sasson, and A. Wigderson, Short proofs are Narrow–Resolution made Simple, *Journal of the ACM*, 48(2), (2001), pp.149-169.
[8] R. Berger, The undecidability of the domino problem, *Mem. Amer. Math. Soc.*, 66, (1966), pp. 1-72.
[9] A. Blake, *Canonical expression in boolean Algebra*, Ph.D Thesis, (1937), University of Chicago.
[10] M.L. Bonet, J.L.Esteban, N. Galesi and J. Johannsen, Exponential separations between Restricted Resolution and Cutting Planes Proof Systems, In *39th Symposium on Foundations of Computer Science*, (FOCS 1998), pp.638-647.
[11] K. Büning, T. Lettman, *Aussangenlogik: Deduktion und Algorithmen*, (1994), B.G Teubner Stuttgart.
[12] E. Burks, *Theory of Self-reproduction*, University of Illinois Press, Chicago, (1966).
[13] S. Buss (ed.), *Handbook of Proof Theory*, North-Holland, (1998).
[14] S. Cavagnetto, Some Applications of Propositional Logic to Cellular Automata, *Mathematical Logic Quarterly*, 55, (2009), pp. 605-616.
[15] M. Clegg, J. Edmonds, and R. Impagliazzo, Using the Groebner basis algorithm to find proofs of unsatisfiability, in *Proceedings of 28th Annual ACM Symposium on Theory of Computing*, (1996), pp. 174-183.
[16] P. Clote and A. Setzer, On PHP st-connectivity and odd charged graphs, in P. Beame and S. Buss, editors,*Proof Complexity and Feasible Arithmetics*, AMS DIMACS Series Vol. 39, (1998), pp. 93-117.
[17] P. Clote and E. Kranakis, Boolean Functions and Computation Models, Texts in Theoretical Computer Science, Springer-Verlag, (2002).
[18] S. Cole, Real-time computation by n-dimensional iterative arrays of finite-state machine, *IEEE Trans. Comput*, C(18), (1969), pp. 349-365.
[19] S. A. Cook, The complexity of theorem-proving procedures, in *Proc. 3rd Ann. ACM Symp. on Theory of Computing*, (1971), pp. 151-158.
[20] S. A. Cook and A. R. Reckhow, On the lengths of proofs in the propositional calculus, in *Prooceedings of the Sixth Annual ACM Symposium on the Theory of Computing*, (1974), pp. 15-22.
[21] S.A Cook and A.R. Reckhow, The relative efficiency of propositional proof systems, *Journal of Symbolic Logic*, 44(1), (1979), pp. 36-50.
[22] S. A. Cook, The P versus NP Problem, *Manuscript prepared for the Clay Mathematics Institute for the Millennium Prize Problems*, (2000).
[23] W. Cook, C. R. Cullard, and G. Turan, On the complexity of cutting planes proofs, *Discrete Applied mathematics*, 18, (1987), pp.25-38.

[24] W. Craig, Three uses of the Herbrand-Gentzen theorem in relating model theory and proof theory, *Journal of Symbolic Logic*, 22(3), (1957), pp. 269-285.
[25] M. Davis, *Computability and Unsolvability, Dover Pubblications*, Inc, New York, (1958).
[26] M. Davis and H. Putnam, A computing procedure for quantification theory, *Journal of the ACM*, 7(3), pp. 210-215.
[27] M. Delorme and J. Mazoyer, editors, *Cellular Automata: a parallel model*, Mathematics and its Application, Springer, (1998).
[28] B. Durand, Inversion of 2D cellular automata: some complexity results, *Theoretical Computer Science*, 134, (1994), pp.387-401.
[29] M. R. Garey and D.S. Johnson, *Computers and Intractability - A guide to the theory of the NP-completeness*, W. H. Freeman, (1979).
[30] A. Goerdt, Cutting planes versus Frege proof systems, in: *Computer Science Logic:4th workshop, CSL '90*, E. borger and et al., eds, Lecture Notes in Computer Science, Spriger-verlag, (1991), pp.174-194.
[31] D. Hilbert and W. Ackermann, *Principles of Mathematical Logic*, New York, (1950).
[32] K. Iwama and S. Miyazaki, Tree-like Resolution is superpolinomially slower then dag-like resolution for the Pigeonhole Principle, in A. Aggarwal and C.P. Rangan, editors, *Proceedings: Algorithms and Computation, 10th International Symposium*, ISAAC'99, Vol. 1741, (1999), pp 133-143.
[33] J. Kari, Reversibility of 2D cellular automata undecidable, *Physica*, D(45), (1990), 379-385.
[34] J. Kari, Reversibility and surjectivity problems of cellular automata, *Jour. Comput. System Scie.*, 48, (1994), pp. 149-182.
[35] J. Kari, Reversible Cellular Automata, *Proceedings of DLT 2005, Developments in Language Theory*, Lecture Notes in Computer Science,3572, pp. 57-68, Springer-Verlag, (2005).
[36] J. Kari, Theory of cellular automata: A survey, *Theoretical Computer Science*, 334, (2005), pp. 3-33.
[37] J. Krajíček, *Bounded arithmetic, propositional logic, and complexity theory*, Encyclopedia of Mathematics and Its Applications, 60, Cambridge University Press, (1995).
[38] J. Krajíček, Propositional proof complexity I., *Lecture notes*, available at http://www.math.cas.cz/krajicek/biblio.html.
[39] J. Krajíček, Dehn function and length of proofs, *International Journal of Algebra and Computation*, 13(5),(2003), pp.527-542.
[40] J. Krajíček, Lower bounds to the size of constant-depth propositional proofs, *Journal of Symbolic Logic*, 59(1), (1994), pp.73-86.
[41] J. Krajíček, Interpolation theorems, lower bounds for proof systems, and indipendence results for bounded arithmetic, *Journal of Symbolic Logic*, 62(2), (1997), pp. 457-486.
[42] L. Levin, Universal search problem (in russian), *Problemy Peredachi Informatsii 9*, (1973), 115-116.
[43] E.F. Moore, Machine models of self-reproduction, *Proc. Symp. Appl. Math. Soc.*, 14, (1962), pp. 13-33.
[44] J. Myhill, The converse to Moore's garden-of-Eden theorem, *Proc. Amer. Math. Soc.*, 14, (1963), pp.685-686.
[45] D. Mundici, NP and Craig's interpolation theorem, *Proc. Logic Colloquium 1982*,North-Holland, (1984), pp. 345-358.
[46] J. von Neumann, The General and Logical Theory of Automata, in *Collected Works*, vol. 5, Pergamon Press, New York, (1963), pp. 288-328.
[47] J. von Neumann, *Theory of Self-reproducting automata*, ed. W. Burks, University of Illinois Press, Chicago, (1966).

[48] J. von Neumann, *Theory of automata: construction, reproduction and homogeneity*, unfinished manuscript edited for pubblication by W. Burks, see (12) pp. 89-250.
[49] C. H. Papadimitriu, *Computational Complexity*, Addison-Wesley, (1994).
[50] C. H. Papadimitriu, NP-completeness: A Retrospective, in *Proceedings of the 24th International Colloquium on Automata, Languages and Programming* 1256, Lecture Notes in Computer Science, Springer, (1997), pp. 2-6.
[51] P. Pudlák, The Lenghts of Proofs, in *Handbook of Proof Theory*, ed. S. Buss, North-Holland, (1998), ch. 8, pp. 547-637.
[52] P. Pudlák, Lower bounds for resolution and cutting plane proofs and monotone computations, *Journal of Symbolic Logic*, (1997), pp. 981-998.
[53] A. A. Razborov, Unprovability of lower bounds on the circuits size in certain fragments of bounded arithmetic, *Izvestiya of the R. A. N.*, 59(1), (1995), pp. 201-224.
[54] D. Richardson, Tesselations with local transformations, *Jour. Comput. System Scie.*, 6,(1972), pp. 373-388.
[55] J. A. Robinson, A machine-oriented logic based on the resolution principle, *Journal of the ACM*, 12(1), pp. 23-41.
[56] M. Sipser, *Introduction to the Theory of Computation*, PWS Publishing Company, Boston, (1997).
[57] M. Sipser, The history and the status of the P versus NP question, *STOC*, (1992), pp. 603-618.
[58] S. Smale, Mathematical problems for the next century, in *Mathematics: Frontiers and perspectives*, AMS, (2000), pp. 271-294.
[59] A. Smith III, A Simple computatio-universal spaces, *Journal of ACM*, (1971), 18, pp. 339-353.
[60] K. Sutner, De Bruijn graphs and linear cellular automata, *Complex Systems*, 5, (1991), 19-31.
[61] T. Toffoli and N. Margolus, *Cellular Automata Machines*, MIT Press, Cambridge MA, (1987).
[62] G. S. Tseitin, On the complexity of derivation in propositional calculus, in A. Slisenko ed., *Studies in Constructive Mathematics and Mathematical Logic*, (1970), Consultants Bureau, New York, pp. 115-125.
[63] A. Turing, On computable numbers with an application to the Enthscheidungsproblem, *Proc. London Math. Soc.*, 42, (1936), pp. 230-265.
[64] A. Urquhart, The Complexity of Propositional Proofs, *Bulletin of Symbolic Logic*, 1(4),(1996), pp. 425-467.
[65] A. Wigderson, P, NP and Mathematics-a computational complexity perspective, http://www.math.ias.edu/avi/BOOKS/.
[66] K. Wagner and G. Wechsung, *Computational Complexity*, Riedel, (1986).

23

Biophysical Modeling using Cellular Automata

Bernhard Pfeifer
University for Medical Informatics and Technology Tyrol
Austria

1. Introduction

The definition of a model is as following: *a model is an abstract and simplified picture of a reality in the world*. From this definition it can be derived that each biomedical model is an approximation of an organism, organ, tissue, or cell. The reason for the need of models is, that the originals are much too complex to be described and understood completely.
Let us assume that an individual has to be completely modeled in a computer system, one has to calculate the information content. This individual should consist of 60 different atoms. Those are defined to their position with accuracy of $\pm 2 \cdot 10^{-12}[m]$. The composition of the atoms is neglected here. Furthermore, the dimension of the individual is given by $2[m] \cdot 0,5[m] \cdot 0,4[m]$. The simplification of position, the neglecting of composition, and the cubic form of the individual is of course a simplification, but let forget about this fact in this example.
Claude Shannon, the founder of the information theory Shannon (1948), defined the entropy **H** of a given information **I** over an alphabet **Z** by

$$H(I) = -\sum_{j=1}^{|Z|} p_j \cdot ld(p_j), \tag{1}$$

where p_j is the likelihood that the j^{th} symbol z_j of the alphabet **Z** appears in the given information text **I**. The unit of entropy is bit. When multiplying H with the number of characters given in the alphabet, the minimum necessary number of bits for representing the information is calculated.
The information content (entropy) of the above given example is calculated as $H = ld(60) = 5,9[bit]$ (Eq. 1). The number of different atom positions in the individual is $V/Vd + 1 = 0.4^3/(64 \cdot 10^{-36}[m^3]) + 1 = 625 \cdot 10^{31} + 1$ [atoms]. Having this it is possible to compute the necessary memory for storing the model: $625 \cdot 10^{31}[atoms] \cdot 5.9[bit/atom] = 3,68 \cdot 10^{34}$. No computer exists that is able to store and handle this model in a finite time.
On account, this simple example shows that using approximations of the reality is necessary to address with complexity for understanding life. Complex problems have to be split into treatable parts, which can then be implemented using modern computers. Such a proceeding is widely used in computer science e.g. when developing algorithms divide and conquer technique are applied for simplifying the problem. This allows computing partial solutions that are then combined to obtain the overall solution.
As it is inevitable to have approximations as models, it seems to be imperative to define the necessary details in the model. Bossel defined a model following: *the scope of the model is the*

key for any model development; its precise specification enables a clear and compact formulation of the model Bossel (1992); Fischer (2006).

2. Modeling techniques

Numerical simulation is typically based on continuous models. Problems, which consider independent variables e.g. spatial elements and time can be modeled using partial differential equations (PDE), and problem without considering more independent variables can be formulated using ordinary differential equations (ODE). Furthermore, cellular automata can be used for modeling spatiotemporal problems.

2.1 Ordinary differntial equations

As one example the Lotka-Volterra-Equation Volterra (1926), which is also know as Predator-Prey-Equation, is a mathematical system based on coupled differential equations, which describe the dynamics of predator and prey populations. The equations are based on three rules, which were constructed during World War II on observations of the fish stock in the Adria. The first rule describes the periodical fluctuations of a population. The fluctuations of the predator-population are phase-delayed compared to the prey-population. The second rule describes the stability of the mean value. Although there are periodical fluctuations, the number of the populations is constant. The last rule describes that the prey-population is growing faster than the predator-population. If predator- and prey-population becomes decimated for a defined period, then the prey-population recovers always faster than the predator-population.
This situation can be mathematical modeled using ordinary differential equations:

$$\frac{dx}{dt} = x(\alpha - \beta y) \tag{2}$$

$$\frac{dy}{dt} = y(\delta x - \gamma), \tag{3}$$

where y is the number of predator-population, x ist the number of prey-population, t represents the time, $\frac{dx}{dt}$ and $\frac{dy}{dt}$ represents the growth of the given populations against time. α is the exponential growing rate of the prey population, β is the predator-capture-rate, δ is the reproduction rate of the predator population, and γ the natural predator death rate.
Models of this type are of importance in theoretical biology, and in epidemiology e.g. for describing the processes of disease spreading.
The disadvantage is, that the result describes the behavior of the whole population without having the chance to look at each individual directly. Another point is, that it is impossible to model topological structures, which may get important for some simulation cases.

2.2 Partial differential equations

As described, using ordinary differential equations enables only to model one dimension. Therefore, in the Predator-Prey-Model the overall behaviors over the time can be modeled and expressed. Partial differential equations, however, consider change of state along several dimensions.
The Predator-Prey-Model can be extended using PDEÕs for modeling spatial variations. These variations are that a predator has to move in order to catch a prey, and a prey is able to move for evading a predator. The model can be described as following:

$$u_t(r,t) = \varphi\nabla^2 u(r,t) + \frac{\alpha}{b}u(b-u) - \gamma\frac{uv}{u+h} \tag{4}$$

$$v_t(r,t) = \psi\nabla^2 v(r,t) + \kappa\gamma\frac{uv}{u+h} - \mu v. \tag{5}$$

The first equation describes the prey population, and the second equation describes the predator population. $\frac{\alpha}{b}u(b-u)$ describes the local growth and mortality of the prey, $\gamma\frac{uv}{u+h}$ describes the predation, κ is the food utilization, and μ the mortality rate of the predator. The parameter α is the maximal growth rate of the prey, b is the carrying capacity of the prey population, and h ist the half-saturation abundance of prey. φ and ψ are the diffusion coefficients.

2.3 Spatiotemporal cellular automata modeling

Cellular automata John von Neumann (1966) are discrete dynamical systems, which allow designing spatiotemporal models based on cell-cell, cell-medium and cell-medium-cell interactions. Cellular models have been introduced by John von Neumann and Stanislaw Ulam as computer models for self-reproduction. The constructed automaton consisted of 29 different states per cell. In the 60ies John Horton Conway created the well known zero player game ÒGame of LifeÓ, which is based on a cellular automaton. Stephen WolframÕs book ÒA New Kind of ScienceÓ Stephen Wolfram (2002) in the year 2002 tried to show how powerful cellular automata are, and that they can be used for modeling in sciences. Based on this book, the public interest increased rapidly and many research laboratories used cellular automaton models for simulating the dynamics of spatiotemporal problems.

2.3.1 Concept behind cellular automata

Cellular automaton models consists of several interacting cells, each of which having a state transition function inside, which brings a cell state at time point t in another state for time point $t+1$. Therefore, a CA model can be anatomized in simple finite state machines, or finite automata.

2.3.1.1 Finite automata

are models for computers for devices with limited resources. A vending machine, but even a computer system we are used to work with comes into this category. Memory and computational resources are limited, but in spite of that, one is able to perform complex operations and simulation using those machines. A deterministic finite automaton can be defined as following

$$A = (Q, \Sigma, \delta, q_0, F), \tag{6}$$

where Q describes the set of states, Σ describes the input alphabet, δ is the state transition function that is defined as $\delta : Q \times \Sigma \to Q$. q_0 is the start state with $q_0 \in Q$. and F represents the set of accepting or final states with $F \subseteq Q$.

2.3.1.2 Finite automata interplay

A cellular automaton consists of a quantum of finite automata, which are collaborating. This collaboration is defined in the state transition function δ where the new state does not only depend on the actual cell states but also from the neighboring cell states. Furthermore, one has to decide, which cells should collaborate. Therefore, the cell adjustment and the neighborship

have to be defined. As each cell also has an output function, the definition of a cellular automaton can be written as follows

$$CA = (C, N, \Sigma, Y, Q, \delta, \sigma, q_0, F), \tag{7}$$

where C is the cell adjustment, N defines the neighborhood, Σ is the input alphabet, Y the output alphabet, and Q describes the set of states. δ is the state transition function with $\delta : Q \times Q^{n_i} \rightarrow Q$ with $n_i \in N$. σ is the output function with $\sigma : Q \rightarrow A$. q_0 is the start state with $q_0 \in Q$, and F represents the set of accepting or final states with $F \subseteq Q$.
As a cellular automaton has a cellular space or lattice, these models allow the visualization of the automaton states at each time point. The regular lattice $L \subseteq R^d$ consists of several individual cells, which interact using a neighborhood relation. The interaction neighborhood can be defined as

$$N_b^I(r) := \{r + c_i | c_i \in N_b^I\} \in L, \tag{8}$$

where N is the interaction neighborhood template, b is the coordinate number, r is the position of the cell and c_i denotes the interacting neighbors.
In two dimensions the only regular polygons forming a regular tesselation are triangles, rectangles and hexagons NY (1977). The neighborhood is defined as:

$$N_b = \{c_i, i = 1, ..., b : c_i = (cos(\frac{2\pi(i-1)}{b}), sin(\frac{2\pi(i-1)}{b}))\}. \tag{9}$$

3. Disease spread modeling using cellular automaton approach

3.1 History

In the year 431 before Christ, Thukydides noted down about a tragedy that devastated Athens citizens. The symptoms were dramatic. Young, healthy adults suddenly came down with an unexplainable disease. This outbreak was the start of an era where epidemics were recorded. Up to the 20ths century contagious diseases ran rampant and often incurable, which dramatically reduced the population in case of an outbreak. From the scores of epidemics occurred in the history, some of them have a special impact up to now: The outbreak came nearly two millennia after the outbreak in Athens, which brought death and bane over the world. Medical historians discovered that the plague occurred in the year 1331 in China, and decimated the population by 50 percent. Over existing trade routes the plague reached Krim in the year 1346, and from this hub Europe, Northern Africa and the Middle East. The name of this disease became the embodiment of horror: Black Death. At that time the disease and the infection was mysterious, but today we know that a bacterium named Yersinia pestis spread using flea living on black rats, infected individuals, and killed between 1347 and 1351 a third of the Europeans back then. Above all the plague bacterium can be transmitted from an infected individual to a healthy individual, which is known as airborne infection. The outbreak changed the behavior of the population in various ways. Some kept themselves away from remaining population to prevent from population contact, while others started to live an extensive life. The next outbreak of the plague appeared in 1896, and spread to nearly every part of the globe. By 1945, the death toll reached approximately 12 million.
Between 1918 and 1920 the Spanish Flu pandemic killed about 20-50 million people, especially young adults and teens with well working immune systems. No infection, no war, and no famine have ever had claimed that much victims in a little while. Surprisingly, the outbreak of the Spanish Flu had no evident impact, because in the heads the scare of the war was present, and nobody wanted to write about this epidemic.

An outbreak of the Asian Flu in 1957 resulted in an estimate of one million deaths. The Hong Kong Flu killed a population of about 700.000 individuals. AIDS, caused by the human immunodeficiency virus (HIV), was first recognized in the 1980s, and it has killed over 20 million people until now. This disease is now a pandemic, with an estimate of more than 40 million infected individuals at present. Apparently there are several factors, which perpetuate the spread of AIDS and other infectious diseases, including incautiousness (both sexually and drug abuse), misconceptions of the transmission and the immense belief in the development of modern medicine. It is worth pointing out in this context that about 90 percent of the death from infectious diseases worldwide is caused by only a few of diseases.

Most contagious diseases can be modeled using mathematical approaches to analyze and understand the epidemiological behavior or for predicting the process. Therefore, different approaches have been developed in the past. The classic S-I-R epidemic model, where class S denotes the number of susceptibles, class I denotes the number of invectives and class R denotes the number of recovered individuals. The sum of the given initial value problem is S(t)+I(t)+R(t)=N, with N being the number of observed population. However, the SIR model is not adequate to model natural birth and death, immigration and emigration, passive immunity and spatial arrangement adequately. To model infection diffusion through space, partial differential equations (PDE) are needed. With PDE models it is possible to simulate the spreading of a disease over a population in space and time. However, the integration of geographical conditions, demographic realities, and keeping track over each individual is impossible. For this purpose, cellular automaton (CA) models can be used. A CA model is a dynamical system in which time and space is discrete and is specified by a regular discrete lattice of cells and boundary conditions, a finite set of cells and states, a defined neighborhood relation, and a state transition function that is responsible for computing the dynamics of the cells over the time.

For this purpose cellular automaton (CA) models can be used. A CA model is a dynamical system in which time and space is discrete and is specified by a regular discrete lattice of cells and boundary conditions, a finite set of cells and states, a defined neighborhood relation, and a state transition function that is responsible for computing the dynamics of the cells over the time. CA models for highly dynamic disease spread simulation are widely known Beauchemin et al. (2004); Castiglione et al. (2007); Xiao et al. (2006) and shape-space interactions were introduced for enabling to simulate complex interacting systems. Dynamic bipartite graphs for modeling physical contact patterns were introduced, which resulted in more precisely modeling of individualsÕ movements. The graph can be built on actual census and available demographic data. When analyzing those graphs, the existing hubs can be found easily. It could be figured out that by using strategies like targeted vaccination combined with early detection without resorting to mass vaccination of a population an outbreak could be contained Eubank et al. (2004). The simulation application EpiSims Barrett et al. (2005), which has been developed at Los Alamos allows simulating different scenarios by modeling the interaction of the different individuals participating in the simulation. The knowledge about the paths enables to perform arrangements like quarantine or targeted vaccination to prevent the disease from further spreading. The model EpiSims was a reproduction of the city Portland (Oregon), but not a facsimile, because to model the habits of about 1,6 million individuals would be nearly impossible and furthermore a massive intrusion into privacy. EpiSims allows to set parameter values for the within-host disease model on demographics of each person, but also simulating the introduction of counter-measures such as quarantine, vaccination or antibiotic use can be done. The human mobility information

is derived from the TRANSIMS model, which estimates the movement of people based on census data and activity maps taken from defined samples of the population. Using this specific information ÒsocialnetworkÓ can be modeled for understanding how epidemiology depends on those characteristics, and furthermore the calculation of the overall economic pecuniary impact is possible.

3.2 Generic disease spread modeling framework

3.2.1 The class `StepResult`

The Class `Stepresult` stores the computed parameters at time point *t* to generate statistics and a snapshot of each individual time point.

```
 1 package Pandemie;
 2
 3 public class StepResult {
 4     private static StepResult instance;
 5
 6    protected long passiveimmunityfrombirth;
 7    protected long susceptible;
 8    protected long infective;
 9    protected long recovered;
10    protected long killedbydisease;
11
12    protected long spontanous;
13    protected long vectored;
14    protected long contact;
15
16    protected long individuals;
17
18    protected long died;
19    protected long born;
20
21    protected long moved;
22    protected long immigrant;
23
24    protected long healty;
25    protected long latent;
26    protected long infectious;
27    protected long removed;
28
29    protected long lastDied;
30    protected long lastKilledDisease;
```

In the attributes (line 4 to 30) the basic computed disease and state values are stored. For accessing these attributes `get` methods are implemented.

```
31    public static synchronized StepResult getInstance() {
32         if (instance == null) {
33           instance = new StepResult();
34         }
35         return instance;
36       }
37
38       protected StepResult() {
39       }
40 }
```

Since the step results need to be accessible in different objects like a global attributes, a singleton pattern Gamma (1994) is used. The instance can be accessed by calling the static `getInstance()` method, which is able to access the protected constructor. Furthermore this method must be synchronized in case multithreading is used to guarantee data consistency.

3.3 The class `InfectionParameters`

The attributes that are managed by this class describe the disease, demographic and action parameters. The initialized values are one set with which it is possible to simulate an avian flu that is highly infective and has a high death rate. Furthermore, parameters for quarantine and medication can be set for simulating different scenarios.

```
 1 package Pandemie;
 2
 3 import java.util.Random;
 4
 5 public class InfectionParameters {
 6    private static InfectionParameters instance;
 7
 8    protected int latentPeriodDays = 3;
 9    protected int infectousPeriodDays = 10;
10    protected int recoveredRemovedAfterDays = 15;
11    protected int incubationPeriodDays = 3;
12    protected int symptomaticPeriodDays = 4;
13
14    protected double birthrate = 0.002d;
15    protected double deathrate = 0.001d;
16
17    protected double virus_morbidity_percent = 0.63d;
18
19    protected double spontaneous_infection_rate = 0.000001d;
20
21    protected double vectored_infection_rate = 0.35d;
22    protected double contact_infection_rate = 0.45d;
23
24    protected double movement_probability = 0.4d;
25
26    protected double immigrantrate = 0.0000001;
27
28    protected boolean useQuarantine = false;
29
30    protected boolean handleMedication = false;
31    protected double medicationOne = 3.5d;
32    protected double medicationTwo = 5.5d;
33
34    protected int suspectibe_again_after_recover = 100;
35    protected int birthimmunityindays = 20;
36
37    protected long maxCellCapacity = 500;
38
39    protected static Random randomGenerator;
40
41    public static synchronized InfectionParameters getInstance() {
42       if (instance == null) {
43          randomGenerator = new Random();
44          instance = new InfectionParameters();
45       }
```

```
46          return instance;
47      }
48
49      protected InfectionParameters() {
50      }
51  }
```

As the access to these parameters is needed by many objects it is also implemented using the singleton pattern.

3.3.1 The class `DiseaseCell`

The `DiseaseCell` class represents one cell of the CA model, in which the individualsÕ takes place and interact among defined rules. The state transition function δ is inherited from the super class. This function is responsible for calculating the spreading, and how infected individuals have to be treated. Therefore, this function calculates death caught by the disease, followed by adding newborns and removing natural death cases. Then, immigrants and emigrants are estimated, the vectored, contact, and the spontaneous infection is computed and in the last step the individual movement is performed.

```
1 package Pandemie;
2
3 public class DiseaseCell extends Cell {
4     private ArrayList<CellIndividual> individuals;
```

In line 4 a dynamical data structure for storing the individuals that take place in the cell is introduced.

```
5     public DiseaseCell(int numberOfIndividuals) {
6         individuals = new ArrayList<CellIndividual>();
7
8         for (int i = 0; i <= numberOfIndividuals - 1; i++) {
9             individuals.add(new CellIndividual());
10        }
11    }
```

When a cell object is instantiated the constructor creates the dynamical data structure that holds the individuals. Furthermore, the number of individuals is generated in the loop and stored using the data structure.

```
12    public boolean performCellAction(CellularAutomaton ca) {
13        this.handleNaturalDeath();
14        this.handleNewborns();
15
16        this.handleImmigrants();
17        this.handleSpontanousInfections();
18
19        this.handleContactInfections(ca);
20        this.handleVectoredInfections();
21
22        this.updateCellIndividuals();
23
24        this.handleMovement(ca);
25        return true;
26    }
```

The state transition function is the main simulation component of the modeling framework. In this implemented model the functions for computing death, birth, spontaneous infections, immigrants, vectored infection contact infection, and individual movement are executed.

```
27    public void updateCellIndividuals() {
28       for (Iterator individualIterator = individuals.iterator();
29          individualIterator.hasNext();) {
30          CellIndividual
31             individual = (CellIndividual) individualIterator.next();
32
33          individual.updateIndividual();
34       }
35    }
```

The `updateIndividual()` method is called in order to initialize the models data for performing the subsequent simulation step over time t.

```
37    public void handleNewborns() {
38       InfectionParameters szParam = InfectionParameters.getInstance();
39       StepResult res = StepResult.getInstance();
40
41       ArrayList<CellIndividual> addIndividuals = new ArrayList<
42             CellIndividual>();
43
44       for (Iterator individualIterator = individuals.iterator();
45          individualIterator.hasNext();) {
46          CellIndividual individual = (CellIndividual) individualIterator.
47                next();
48
49          if ((individual.getAgeType() == AgeType.ADULT)
50                || (individual.getAgeType() == AgeType.TEEN)) {
51             if (isTheCase(szParam.getBirthrate())) {
52                CellIndividual newborn = new CellIndividual();
53                newborn.setAgeType(AgeType.KID);
54                 newborn.setSusceptibleInDays(szParam.
55                         getBirthimmunityindays());
56
57                if (this.isTheCase(0.7d)) {
58                    newborn.setStateType(StateType.PASSIVEIMMUNEFROMBIRTH);
59                }
60
61                addIndividuals.add(newborn);
62                res.setBorn(res.getBorn() + 1);
63             }
64          }
65       }
66       if (addIndividuals != null)
67          individuals.addAll(addIndividuals);
68    }
69
70    public void handleNaturalDeath() {
71       InfectionParameters szParam = InfectionParameters.getInstance();
72
73       for (Iterator individualIterator = individuals.iterator();
74          individualIterator.hasNext();) {
75          CellIndividual individual = (CellIndividual) individualIterator.
```

```
76                  next();
77
78          if (individual.getStateType() == StateType.DIED) continue;
79          if (individual.getStateType() == StateType.KILLEDBYDISEASE)
80                      continue;
81
82          if (this.isTheCase(szParam.getDeathrate())) {
83             if ((individual.getAgeType() == AgeType.KID)
84                || (individual.getAgeType() == AgeType.TEEN)
85                || (individual.getAgeType() == AgeType.ADULT)) {
86                if (this.isTheCase(0.7d)) {
87                    individual.setStateType(StateType.DIED);
88                }
89             } else {
90                 individual.setStateType(StateType.DIED);
91             }
92          }
93       }
94    }
```

Both methods `handleNewborns()` and `handleNaturalDeath()` implements the natural growing and shrinking of a population caused by defined birth and death parameters. When an individual gets is born a temporary immunity is applied, which protects the individual from becoming ill by the spreading disease. Furthermore, in this model it is only possible for adults to get children, which is accordable with natural behavior. During the computation of natural death cases a stochastic function is used, which gives the different age classes (kids, teen, adult, elderly) a different likelihood of dieing.

```
95    public void handleImmigrants () {
96       InfectionParameters szParam = InfectionParameters.getInstance();
97       StepResult res = StepResult.getInstance();
98
99       if (this.isTheCase(szParam.getImmigrantrate())) {
100         CellIndividual immigrant = new CellIndividual ();
101         if (immigrant.getAgeType() == AgeType.KID)
102            immigrant.setAgeType(AgeType.ADULT);
103
104          individuals.add(immigrant);
105          res.setImmigrant(res.getImmigrant() + 1);
106      }
107   }
```

Defined by the immigration rate parameter the probability of a new immigrant is computed. If the function returns that a new immigrant is allowed to enter the simulation then the immigrant is added to the cell as new member. Furthermore, there is a restriction that only adults and elderly people are allowed to enter. If a individual not being part of this age type tries to enter, then the age class is adapted in order to fulfill the requirements.

```
108   protected void handleNeighborCellInfections (
109      CellularAutomaton ca, InfectionParameters szParam,
110      StepResult res, double probability) {
111      DiseaseCell regSZNeighbourCell;
112
113      for (Iterator individualIterator = individuals.iterator();
```

```
114          individualIterator.hasNext();) {
115          CellIndividual individual = (CellIndividual) individualIterator.
116                next();
117
118          if (szParam.isUseQuarantine() &&
119             (individual.getQuarantineType() == QuarantineType.QUARANTINE))
120             continue;
121
122          for (Iterator it = neighbourCellIndexList.iterator();
123             it.hasNext();) {
124             Long element = (Long) it.next();
125             try {
126                regSZNeighbourCell = (DiseaseCell)
127                   ca.getCell(element);
128             } catch (Exception e) {
129                continue;
130             }
131
132             for (Iterator adjacentIndividual = regSZNeighbourCell.
133                 getIndividuals().iterator();
134                adjacentIndividual.hasNext();) {
135                CellIndividual adjacent = (CellIndividual)
136                     adjacentIndividual.next();
137
138                if (szParam.isUseQuarantine() &&
139                   (individual.getQuarantineType() == QuarantineType.
140                     QUARANTINE))
141                   continue;
142
143
144                if (individual.getStateType() == StateType.INFECTIVE) {
145                   switch (adjacent.getStateType()) {
146                   case SUSCEPTIBLE:
147                      boolean infection = this.isTheCase(probability);
148                      if (adjacent.getSusceptibleInDays() > 0) infection =
149                            false;
150                      if (individual.getDiseaseCycle() ==     DiseaseCycle.
151                            LATENT)
152                         infection = false;
153
154                      if (infection) {
155                          adjacent.setStateType(StateType.INFECTIVE);
156                          adjacent.setDiseaseCycle(DiseaseCycle.LATENT);
157
158                          adjacent.setInfectedSinceDays(1);
159                          res.setContact(res.getContact() + 1);
160                      }
161                      break;
162                   }
163                }
164             }
165          }
166       }
167    }
168
169       protected void handleSameCellInfections (
170          StepResult res, double probability, boolean contactInfection) {
171       ...
```

```
172     }
```

Based on the given neighborhood relation the individuals in the cells do have a likelihood to interact. The methods `handleNeighborCellInfections()` and `handleSameCellInfections()` are responsible for computing these connection probabilities. Furthermore, when two individuals are contacting and one of them is suffering from the disease, the infection probability is computed and the individualÕs parameters are set. Due to the reason that the methods are quite similar the more complex ones code is depicted (line 108-177).

```
173     public void handleContactInfections(CellularAutomaton ca) {
174        InfectionParameters szParam = InfectionParameters.getInstance();
175        StepResult res = StepResult.getInstance();
176
177        handleSameCellInfections (res, szParam.getContact_infection_rate(),
178              true);
179        handleNeighborCellInfections (ca, szParam, res,
180           szParam.getContact_infection_rate());
181     }
182
183     public void handleVectoredInfections() {
184        InfectionParameters szParam = InfectionParameters.getInstance();
185        StepResult res = StepResult.getInstance();
186
187        handleSameCellInfections(res, szParam.getVectored_infection_rate(),
188              false);
189     }
```

The state transition function δ computes the so-called vectored infections and the contact infections. Thus the methods `handleContactInfections()` and `handleVectoredInfections()` exists, which are using the helper methods `handleNeighborCellInfections()` and `handleSameCellInfections()` described above.

```
190     public void handleSpontanousInfections () {
191        InfectionParameters szParam = InfectionParameters.getInstance();
192        StepResult res = StepResult.getInstance();
193
194        for (Iterator individualIterator = individuals.iterator();
195           individualIterator.hasNext ();) {
196           CellIndividual individual = (CellIndividual) individualIterator.
197                 next ();
198
199           if (this.isTheCase(szParam.getSpontaneous_infection_rate())) {
200              individual.setStateType(StateType.INFECTIVE);
201              individual.setDiseaseCycle(DiseaseCycle.LATENT);
202              individual.setInfectedSinceDays(1);
203              res.setSpontanous(res.getSpontanous()+1);
204           }
205        }
206     }
```

If spontaneous infection is turned on in the simulation parameters are used for computing a probability if a spontaneous infection occurs at the actual time point at the actual individual.

```
207 public void handleMovement(CellularAutomaton ca) {
208    InfectionParameters szParam = InfectionParameters.getInstance();
209    StepResult res = StepResult.getInstance();
210
211    long index = 0;
212    int whereToMove = 0;
213    int ctr;
214    DiseaseCell regSZNeighbourCell;
215
216    int cellMembers = this.individuals.size();
217    for (int cellNumber = 0; cellNumber <= cellMembers - 1; cellNumber++) {
218       if ((this.isTheCase(szParam.getMovement_probability())) &&
219          (this.individuals.size() > 0)) {
220          whereToMove =
221              InfectionParameters.randomGenerator.nextInt(
222             this.getNeighbours().size());
223
224          Iterator findIterator = this.getNeighbours().iterator();
225          ctr = 0;
226          while (findIterator.hasNext()) {
227             if (ctr >= whereToMove) break;
228             try {
229                regSZNeighbourCell = (DiseaseCell) ca.getCell((Long)
230                         findIterator.next());
231                index = regSZNeighbourCell.cellIndex;
232             } catch (Exception e) { }
233             ctr++;
234          }
235
236          try {
237             DiseaseCell newCellPosition = (DiseaseCell) ca.getCell(index);
238             if ((newCellPosition.individuals.size() < szParam.
239                   getMaxCellCapacity ())) {
240                if ((newCellPosition != null) && (newCellPosition.
241                         individuals != null)) {
242                   CellIndividual individuum =    this.individuals.get(0);
243                   ArrayList<CellIndividual> copyIndividuals = new ArrayList
244                         <CellIndividual>();
245                   ArrayList<CellIndividual> newIndividuals = new ArrayList
246                         <CellIndividual>();
247                    newIndividuals.addAll(newCellPosition.individuals);
248                    newIndividuals.add(individuum);
249                   copyIndividuals.addAll(1, this.individuals);
250                   this.individuals.clear();
251                   this.individuals = copyIndividuals;
252                    newCellPosition.individuals.clear();
253                   newCellPosition.individuals = newIndividuals;
254
255                   res.setMoved(res.getMoved() + 1);
256                }
257             }
258          } catch (Exception e) {}
259       }
260    }
261 }
```

The method `handleMovement()` computes using a random number if and where the individuals of the cell should move. Moving paths are strictly limited to the underlying

neighborhood relation. As expanded neighborhoods can be defined, it is possible that one individual can move long distances in one single time step. To give one example, using such a neighborhood relation enables to connect far-off places connected by infrastructur circumstances like airports. These far distance neighbors can be disconnected during the simulation, as airports were closed in China during the SARS outbreak.

3.3.2 The class `Cellindividual`

Each cell is able to hold a set of individuals, and furthermore, each individual has another finite state automaton working inside. Thus, it is possible to store the actual state and actual parameters of each individual. Using these parameters it is possible to control each individual separately. For example, it is possible to set quarantine parameters for some individuals or to use a special medication. These lists are also known as meme lists.

```
1 package Pandemie;
2
3 public class CellIndividual {
4     public enum StateType {
5         PASSIVEIMMUNEFROMBIRTH, SUSCEPTIBLE,
6         INFECTIVE, RECOVERED, KILLEDBYDISEASE, DIED
7     }
8
9     public enum AgeType {
10        KID, TEEN, ADULT, ELDERLY
11    }
12
13    public enum TreatmentType {
14        MEDICAL1, MEDICAL2, NOTREATMENT
15    }
16
17    public enum QuarantineType {
18        NORMAL, QUARANTINE
19    }
20
21    public enum DiseaseCycle {
22        HEALTHY, LATENT, INFECTIOUS, REMOVED, NIL
23    }
```

The class `CellIndividual` stores the memes and the different states of each individual. This class allows to model and extend any meme list for simulating social behavior more precicely.

```
24    private StateType stateType;
25    private AgeType ageType;
26    private QuarantineType quarantineType;
27    private TreatmentType treatmentType;
28    private DiseaseCycle diseaseCycle;
29
30    private int infectedSinceDays;
31    private int susceptibleInDays;
32    private double mortalityRateFactor = 1d;
```

Here, the representation of the individual states is implemented. The attributes have to be used for storing the individual memes and states.

```
33    public CellIndividual () {
34       this.setStateType(StateType.SUSCEPTIBLE);
35        this.setTreatmentType(TreatmentType.NOTREATMENT);
36       this.setDiseaseCycle(DiseaseCycle.HEALTHY);
37       this.setInfectedSinceDays(0);
38       this.setSusceptibleInDays(0);
39
40       int ageClass = InfectionParameters.randomGenerator.nextInt(4);
41       switch (ageClass) {
42         case 0 : this.setAgeType(AgeType.KID); break;
43         case 1 : this.setAgeType(AgeType.TEEN); break;
44         case 2 : this.setAgeType(AgeType.ADULT); break;
45         case 3 : this.setAgeType(AgeType.ELDERLY); break;
46         default : this.setAgeType(AgeType.ADULT); break;
47       }
48    }
```

When a new individual is generated the constrcutor must be used. Per definition a new individual is always in state *healthy* and *susceptible*, but using the `set` methods these parameters can be changed. The used age-type is dependent on a random number ranged form [1..4].

```
49    protected double computeMortalityRate (double morbidityValue,
50         InfectionParameters simParam) {
51       double value = 1.0d;
52
53       if (simParam.isHandleMedication()) {
54          value = this.isTheCase(0.5d) ?
55             simParam.getMedicationOne() : simParam.getMedicationTwo();
56       }
57
58       return morbidityValue / value;
59    }
```

The method `computeMortalityRate()` computes the probability of an individual to be killed by the disease dependent on given parameters available in the meme list.

```
60    protected void updateStateType (InfectionParameters simParam) {
61       switch (stateType)
62       {
63         case PASSIVEIMMUNEFROMBIRTH:
64            this.setSusceptibleInDays(this.getSusceptibleInDays()-1);
65            if (this.getSusceptibleInDays() < 1) {
66               this.setSusceptibleInDays(0);
67               this.setInfectedSinceDays(0);
68
69               this.setStateType(StateType.SUSCEPTIBLE);
70               this.setDiseaseCycle(DiseaseCycle.HEALTHY);
71            }
72          break;
73         case SUSCEPTIBLE:
74            this.setInfectedSinceDays(0);
75            this.setSusceptibleInDays(0);
76            this.setDiseaseCycle(DiseaseCycle.HEALTHY);
77            break;
78         case INFECTIVE:
```

```
79              this.setInfectedSinceDays(this.getInfectedSinceDays()+1);
80
81              if (this.getInfectedSinceDays() >=
82                simParam.getRecoveredRemovedAfterDays()) {
83                 double mortalityRate = computeMortalityRate
84                       (simParam.getVirus_morbidity_percent(), simParam);
85
86                 if (this.isTheCase(mortalityRate)) {
87                    this.setStateType(StateType.KILLEDBYDISEASE);
88                    this.setDiseaseCycle(DiseaseCycle.REMOVED);
89                 } else {
90                    this.setStateType(StateType.RECOVERED);
91                    this.setDiseaseCycle(DiseaseCycle.HEALTHY);
92                    this.setInfectedSinceDays(0);
93                    this.setSusceptibleInDays(
94                      simParam.getSuspectibe_again_after_recover());
95                 }
96              }
97              break;
98           case RECOVERED:
99              this.setSusceptibleInDays(this.getSusceptibleInDays()-1);
100             if (this.getSusceptibleInDays() < 1) {
101                this.setStateType(StateType.SUSCEPTIBLE);
102                this.setDiseaseCycle(DiseaseCycle.HEALTHY);
103             }
104             break;
105          case KILLEDBYDISEASE:
106             this.setDiseaseCycle(DiseaseCycle.REMOVED);
107             break;
108          case DIED:
109             this.setDiseaseCycle(DiseaseCycle.NIL);
110             break;
111       }
112    }
```

This functions is for updating the individuals state. The parameters are stored in the singleton object, which holds the data of the disease being simulated.

```
113    protected void updateDiseaseCycle (InfectionParameters simParam) {
114       switch (diseaseCycle)
115       {
116          case HEALTHY:
117            break;
118          case LATENT:
119             if (this.getInfectedSinceDays() > simParam.getLatentPeriodDays
120                   ())
121                this.setDiseaseCycle(DiseaseCycle.INFECTIOUS);
122             break;
123          case INFECTIOUS:
124             if (this.getStateType() == StateType.KILLEDBYDISEASE)
125                this.setDiseaseCycle(DiseaseCycle.REMOVED);
126
127             if (this.getStateType() == StateType.RECOVERED)
128                this.setDiseaseCycle(DiseaseCycle.HEALTHY);
129             break;
130          case REMOVED:
131             this.setDiseaseCycle(DiseaseCycle.NIL);
```

```
132             break;
133         }
134     }
```

This functions is for updating the individuals disease life cycle state. The parameters are also stored in the singleton object, which holds the data of the disease being simulated.

```
135     public void updateIndividual () {
136         InfectionParameters simParam = InfectionParameters.getInstance();
137         StepResult sRes = StepResult.getInstance();
138 
139         updateStateType (simParam);
140         updateDiseaseCycle (simParam);
141 
142         if (simParam.isUseQuarantine()) checkQuarantine ();
143 
144         adaptStatistics (sRes);
145     }
```

Each individual state needs to be updated after a simulation step. The method `updateIndividual()` handles this and calls a method for updating the step and individual statistics for performing analysis afterwards.

```
146     public void adaptStatistics (StepResult sRes) {
147         switch (stateType)
148         {
149           case PASSIVEIMMUNEFROMBIRTH:
150               sRes.setPassiveimmunityfrombirth(
151                sRes.getPassiveimmunityfrombirth()+1);
152            break;
153           case SUSCEPTIBLE:
154               sRes.setSusceptible(sRes.getSusceptible()+1);
155               break;
156           case INFECTIVE:
157               sRes.setInfective(sRes.getInfective()+1);
158               break;
159           case RECOVERED:
160               sRes.setRecovered(sRes.getRecovered()+1);
161               break;
162           case KILLEDBYDISEASE:
163               sRes.setKilledbydisease(sRes.getKilledbydisease()+1);
164               break;
165           case DIED:
166               sRes.setDied(sRes.getDied()+1);
167         }
168 
169         switch (diseaseCycle)
170         {
171           case HEALTHY:
172               sRes.setHealty(sRes.getHealty()+1);
173               break;
174           case LATENT:
175               sRes.setLatent(sRes.getLatent()+1);
176               break;
177           case INFECTIOUS:
178               sRes.setInfectious(sRes.getInfectious()+1);
```

```
179             break;
180           case REMOVED:
181             sRes.setRemoved(sRes.getRemoved()+1);
182             break;
183           case NIL:
184             sRes.setRemoved(sRes.getRemoved()+1);
185             break;
186         }
187     }
```

Updates the general statistics data after each simulation steps.

```
188     public void checkQuarantine () {
189         InfectionParameters simParam = InfectionParameters.getInstance();
190         switch (stateType)
191         {
192         case INFECTIVE:
193           if (this.getInfectedSinceDays() > simParam.getIncubationPeriodDays
194               ()) {
195             this.quarantineType = QuarantineType.QUARANTINE;
196           } else this.quarantineType = QuarantineType.QUARANTINE;
197           break;
198          default: this.quarantineType = QuarantineType.QUARANTINE;
199         }
200     }
201 }
```

The method `checkQuarantine()` is used by the state transition function in case the quarantine option is enabled. If an individual is infected, if the individual shows symptoms, and if quarantine is enabled then the individual is set to quarantine. In this case the individual has no, or a very limited chance, to infect a healthy individual.

3.4 The class `DiseaseSpreadCellularAutomaton`

```
1 package Pandemie;
2
3 public class DiseaseSpreadCellularAutomaton extends CellularAutomaton {
4     public static int timers = 0;
5
6     public void compute() {
7         StepResult sRes = StepResult.getInstance();
8         InfectionParameters simParam = InfectionParameters.getInstance();
9
10        long timer;
11        System.out.println (sRes.getHeader());
12
13        for (timer = this.getStartTime(); timer <= this.getStopTime();
14              timer++) {
15            System.out.print(timer + "\t");
16
17            super.compute();
18
19            this.writeSpread("individuals", false);
20            this.writeSpread("susceptible", false);
21            this.writeSpread("infected", false);
22            this.writeSpread("recovered", false);
```

```
23          this.writeSpread("combined", true);
24
25          adaptParameters(timer, 15, true, false, simParam,
26             sRes, 1.2d, 1.5d, 1.1d, 1.05d);
27          useQuarantineAfter (timer, 50, false, simParam);
28       }
29    }
```

The class `DiseaseSpreadCellularAutomaton` is inherited from the basic class named `CellularAutomaton`. The function of the `compute()` method is to iterate through the CA cells and calls the state transition function δ. Therefore, the method iterates from the start timer to the end point and calls the compute method of the super class. The super class itself calls the method `performAction()`, which is known as the state transition function δ. Furthermore, using the helper method `writeSpread()` the simulation step data is persistently stored, and the method `adaptParameters()` is used for adapting the social behavior and the contact probability. The method `useQuarantineAfter()` could be used for drastic intervention into the system - the usage of quarantine can be enabled and parametrized.

```
30    public long countIndividuals() {
31       long individuals = 0;
32       for (Iterator cellIterator = this.getCellList().iterator();
33          cellIterator.hasNext ();) {
34          DiseaseCell cell = (DiseaseCell) cellIterator.next();
35
36          for (Iterator individualIterator = cell.getIndividuals().
37                iterator();
38             individualIterator.hasNext();) {
39             CellIndividual indiv =
40                (CellIndividual) individualIterator.next();
41
42             if ((indiv.getStateType() != StateType.DIED) &&
43                (indiv.getStateType() != StateType.KILLEDBYDISEASE))
44                individuals++;
45          }
46       }
47
48       return individuals;
49    }
50
51    public void useQuarantineAfter (long timer, int time, boolean doIt,
52       InfectionParameters simParam) {
53       if ((timer >= time) && (doIt))
54          simParam.setUseQuarantine(true);
55    }
56
57    public void adaptParameters(long timer, long reduceAfter, boolean doIt,
58       boolean stopSpontanous, InfectionParameters simParam,
59       StepResult sRes, double reduceSpontanousFactor,
60       double reduceMorbidityFactor, double reduceContactFactor,
61       double reduceVectoredFactor) {
62       if ((stopSpontanous) && (sRes.getInfective() > 0))
63             simParam.setSpontaneous_infection_rate(0.0d);
64
65       if ((doIt) && ((timer % reduceAfter) == 0)) {
66          if (sRes.getInfective() > 0)
67             simParam.setSpontaneous_infection_rate(
```

```
68                  simParam.getSpontaneous_infection_rate() /
69                  reduceSpontanousFactor);
70
71              simParam.setVirus_morbidity_percent(
72                  simParam.getVirus_morbidity_percent() / reduceMorbidityFactor);
73              simParam.setContact_infection_rate(
74                  simParam.getContact_infection_rate() / reduceContactFactor);
75              simParam.setVectored_infection_rate(
76                  simParam.getVectored_infection_rate() / reduceVectoredFactor);
77          }
78      }
79 }
```

3.5 Sample of a virus disease spread simulation

3.5.1 Geographic model

3.5.1.1 Austria

In the first simulation scenario a map of Austria was used. The model was simplified due to a homogenous population density over the whole country. The used map is depicted in figure 1.

Fig. 1. Geographical map of Austria with its nine states.

3.5.1.2 Tyrol

For the second simulation scenario the state Tyrol was chosen. Tyrol has 660.000 inhabitants, where about 115.000 inhabitants are living in the capital Innsbruck. The total area is 10.628 square kilometers. The area of settlement is about 1.600 square kilometers. Figure 2 depicts the geographical map of Tyrol and the population density is figured using colors from white, light yellow up to red.

3.5.1.3 Parameters

The simulated infectious disease used for the simulation is similar to the avian flu, except for the imperative difference that this virtual virus can be transmitted between human beings directly with a relatively high likelihood. Therefore, this virtual form of the H5N1 avian flu virus can be considered a dangerous mutation, which could have the power to effect an epidemic/pandemic situation. Table 2 depicts the parameters that have been used for the simulation experiments.

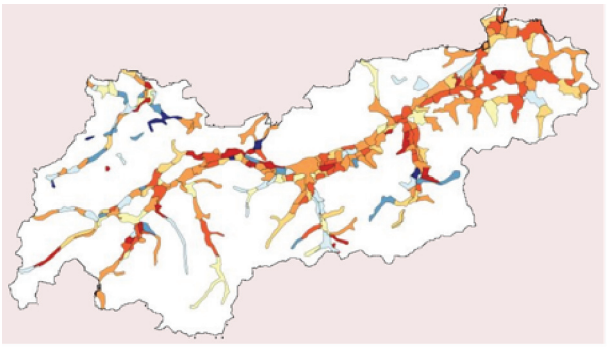


Fig. 2. population density of state Tyrol. The used colors (from white to red) for the population densities specify the density steps from 0, 200, 400, 600, 800 and 1000 inhabitants per square kilometer. The color light gray was used to describe the non-state area.

Table 1. default

Description	Value
Latent period in days	3
Infectious period in days	10
Recovered or removed after days	15
incubation period in days	3
Symptomatic period in days	4
Natural birth rate in percent	0.002
Natural death rate in percent	0.001
Virus morbidity in percent	0.63
Spontaneous infection rate in percent	0.00001
Vectored infection rate in percent	0.4
Contact infection rate in percent	0.6
Movement probability in percent	0.4
Immigration rate in percent	0.0000001
Re-Susceptible (temporary immunity) after days	100
Temporary immunity after birth in days	20

Table 2. Different parameters that were used during the simulation. After 100 time steps, the temporary immunity (Re-Susceptible after days) is lost completely. The parameter values for the infection cycle and the virus morbidity were chosen from the knowledge about the H5N1 human infections. The infection rate (vectored and contact) is supposed to be high in order to simulate a very aggressive (mutated) form of the virus that easily spreads from one individual to another. The other parameters were taken to model the behavior of the state Tyrol best possible.

3.5.2 State transition function δ

The algorithm iterates through each cell of the CA. Each cell represents a small area of the used geographical map and performs the operations of the n individuals placed in the cell (=location). The above described method `performCellAction()` computes the next discrete time step by considering following steps:

1. Handle the natural death cases
2. Handle the natural birth cases
3. Compute death caused by the disease
4. Compute the immigrants
5. Compute vectored infections
6. Compute contact infections
7. Compute spontaneous infections
8. Handle recovered individuals
9. Handle re-susceptible
10. Perform movement operations of the individuals
11. Adapt parameters according specification
12. Create output for actual time step

The steps (1-11) are performed until the specified number of time steps for the simulation is reached. During the simulation process snapshots of the actual distributions are created and furthermore, the data for subsequent statistical analysis is generated and stored. With this information it is possible to track each individual and to reconstruct the occured interactions. This enables the usage of statistical approaches for better understanding the disease spread mechanisms and to identify the best possible way to stop the spreading.

3.5.3 Simulation results

3.5.3.1 Austria

Three scenarios were simulated. The infection seed point was set to the capital of Austria, Vienna. In scenario A, neither medical treatment was provided nor was quarantine declared. In scenario B, two different medications were used for the treatment, but quarantine was not considered. The medication was aimed at increasing the healing chances by 45-55 percent. In scenario C, individuals were submitted to both, medical treatment and quarantine. Furthermore, the social behavior changes of the individuals during the simulation was considered. These behaviors werde modeled because when a disease is circulating, individuals are very cautious contacting others to minimize their own risk of infection.
Figure 3 depicts the development of the susceptibles over the time. As expected from declaring a quarantine status in scenario C, the infection spread stops.
Figure 4 shows the characteristics of the infection over the time in percent and in Figure 5, the fatal cases are illustrated. Assuming that there is no medication, and no quarantine declared, the highest death toll is observed. The difference between scenario B and C is based on the fact that in scenario B the medication is given from the first day on, whereas in scenario C the medication and the quarantine start 50 days after the outbreak.

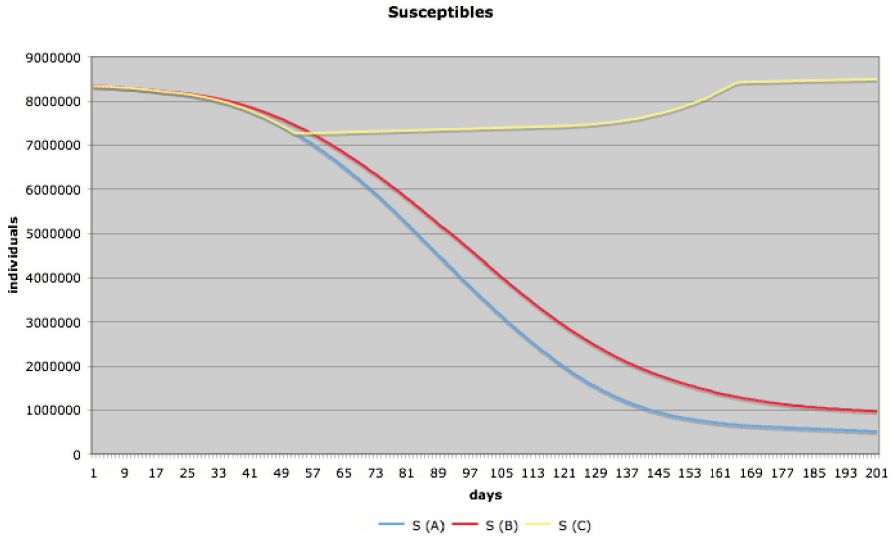


Fig. 3. Susceptible individuals over the simulated period.

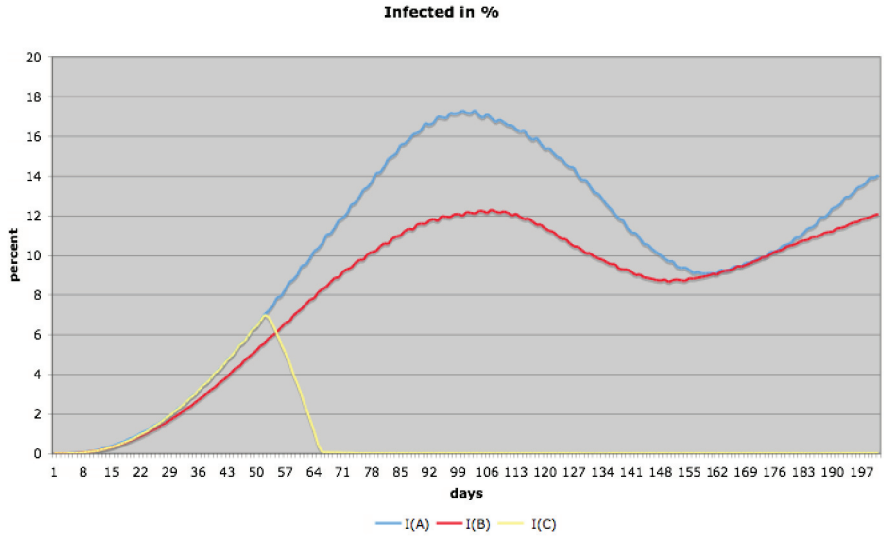


Fig. 4. Development of the infection over the simulated period.

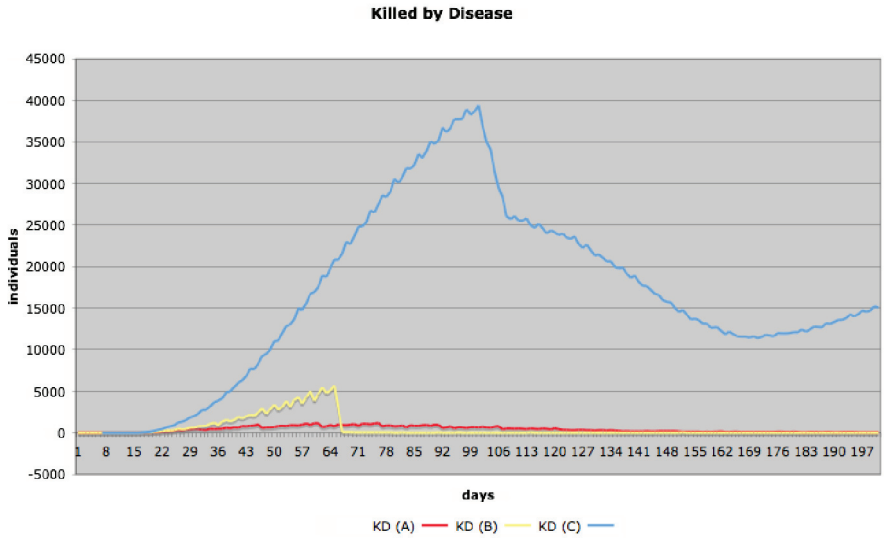


Fig. 5. Fatal cases of the three simulated scenarios.

Figure 6 depicts the parameters individuals, susceptible, infected and removed. As presented in figure 4b, the medi-cation slows down the spreading and reduces the fatal cases dramatically. When quarantine is consistently applied, the spread is controlled after a few days.

Figure 7 depicts the spatial results of the scenarios A, B, C at time point 50 days after outbreak. The dots and grey surfaces depict the areas where infected individuals are located.

At time point 65 days after outbreak (figure 8) the difference between the three simulated scenarios can be seen clearly. When no treatment and no quarantine are applied, the infection spreads the most. The enacted quarantine (C) was able to stop the disease from further spreading few days, the fatal cases were also reduced in scenario B but the disease was still spreading.

3.5.3.2 Tyrol

Eight different scenarios were simulated 4. The seed point of the infection was set to the capital Innsbruck. In the first scenario (scenario A), the disease spread in the state Tyrol where medical treatment was performed. Two different drugs are available for infected individuals. Drug one reduces the death rate by 55 percent, whereas drug two reduces the death rate by 45 percent. The social behavior of the individuals changes during the simulation time, which would also occur in a real situation. When a fatal disease is circulating, individuals are very cautious contacting others to minimize their infection risk. The second scenario (scenario B) is similar to scenario A with the difference that no medical treatment is performed. Scenario C and D is equal to A and B with the difference that there is no adaptation of the social behavior. Scenario E and F is equal to scenario A and B with the difference that after 50 time steps a strictly controlled quarantine is introduced. In the last two scenarios (An, Bn), the same simulation parameters were applied as in A and B with the difference that no geographical and population density was used. Therefore, each cell covers the mean number of individuals from

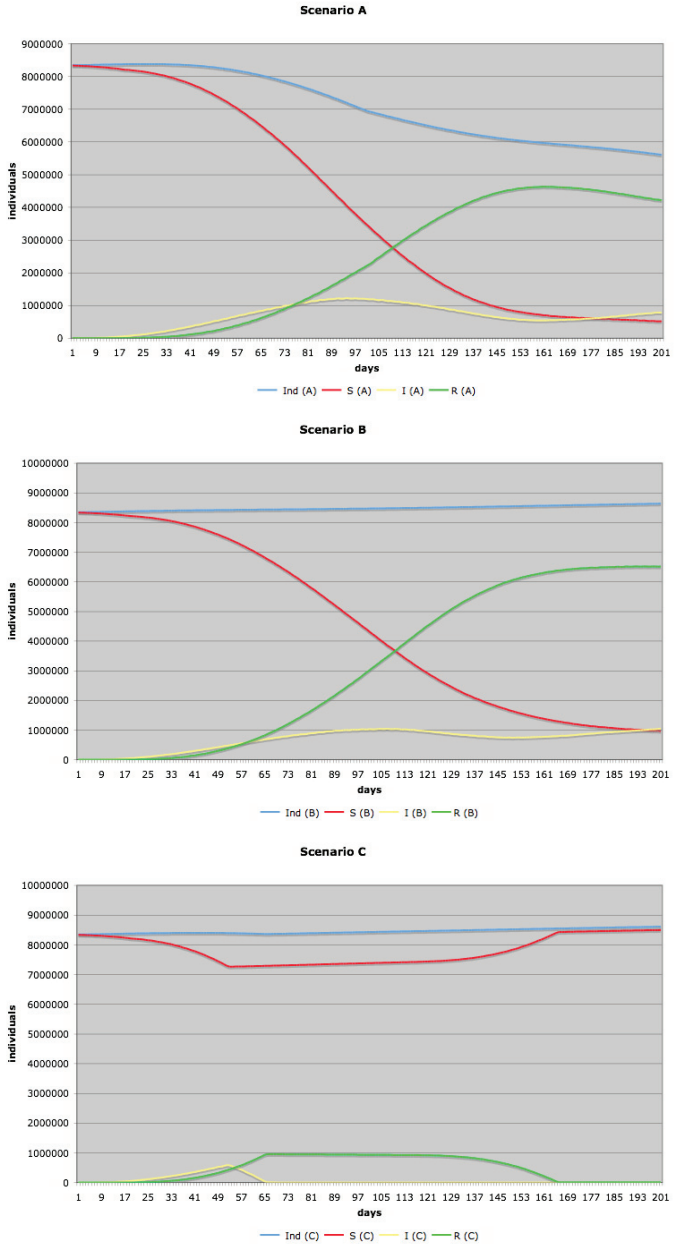


Fig. 6. Population, susceptibles, infected and removed individuals over the simulated period.

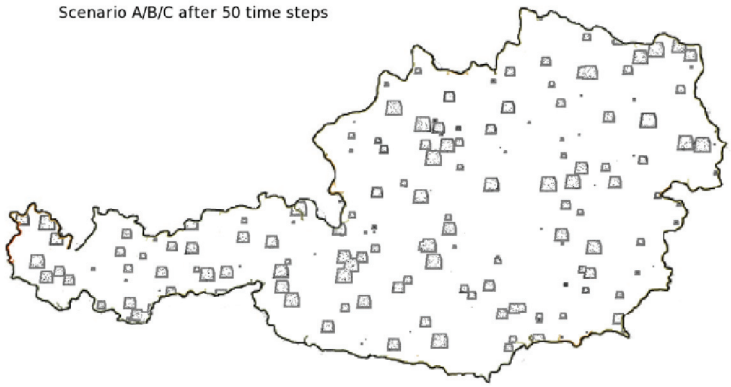


Fig. 7. Screenshot of the spatial result for scenarios A, B, C 50 days after outbreak.

the state Tyrol model. By comparing these scenarios with A and B, it is possible to find out the relevance of geographical (natural barriers) and population density information. The blue color (Ind) is used for the population, red color (S) depicts the susceptible individuals, yellow (I) is used to visualize the infected individuals and the green color (R) was taken to depict the removed or temporarily immune individuals. The virus's transmissibility (R_0 value) is such that each infectious case gives rise to 3.4 secondary infectious cases. The following figures (from figure 9 to figure 16) depict the classes susceptible (S), infected (I), and removed (R).

Table 3. default

scenario	medication	quarantine	social behavior	geographical conditions
A	x		x	x
B			x	x
C	x			x
D				x
E	x	x	x	x
F		x	x	x
An	x		x	
Bn			x	

Table 4. Overview of the different simulation scenarios in tabular view (x stands for true, no character for false). For more information see text.

In figure 17, the changes in population over the time are depicted. Figure 18 shows the fatal cases caused by the disease aggregated per month.
In the simulation, the value for the natural birth rate was 0.002 and the natural death rate 0.001. An infected individual can be removed during the simulation for three different reasons. The first way is that the individual is removed because of natural death, and then the individual can be removed because the disease ended fatal and the third way to be removed to another class is that the individual got healthy again. Figure 19 shows the percentage between natural

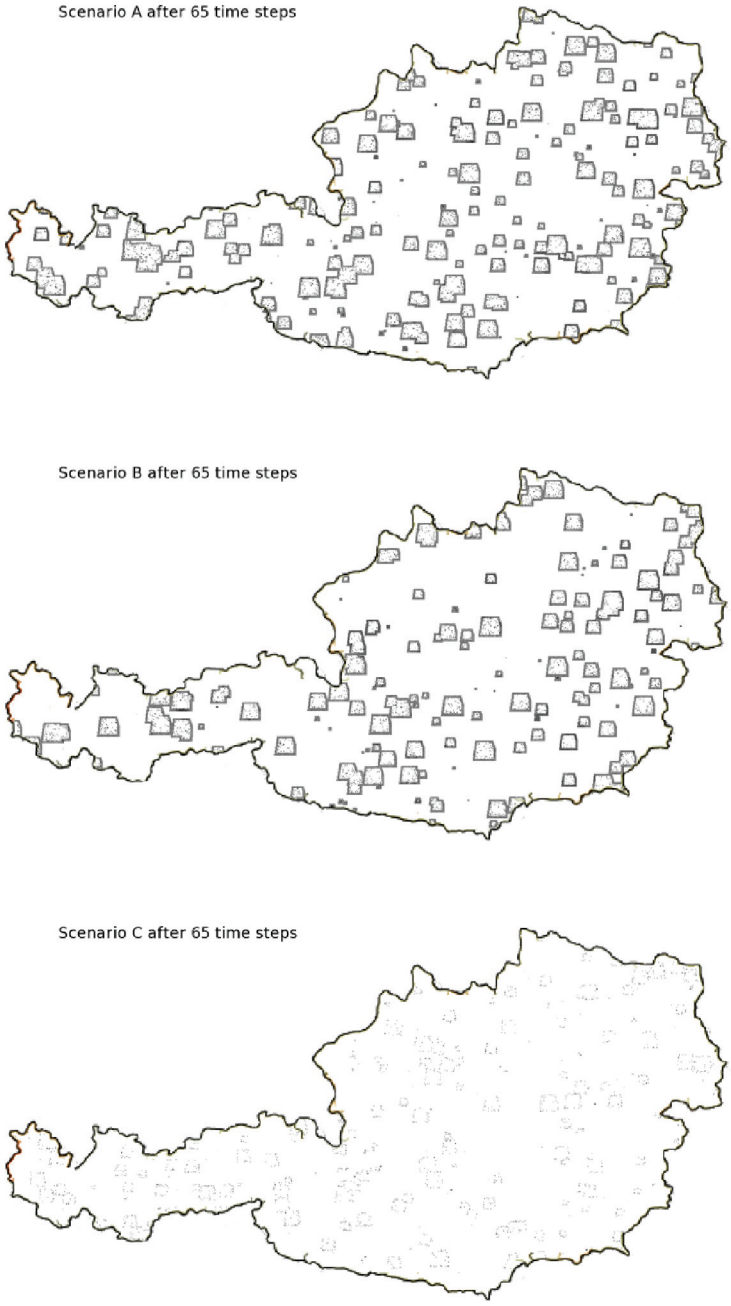


Fig. 8. Screenshot of the spatial result for scenarios A, B, C 65 days after outbreak.

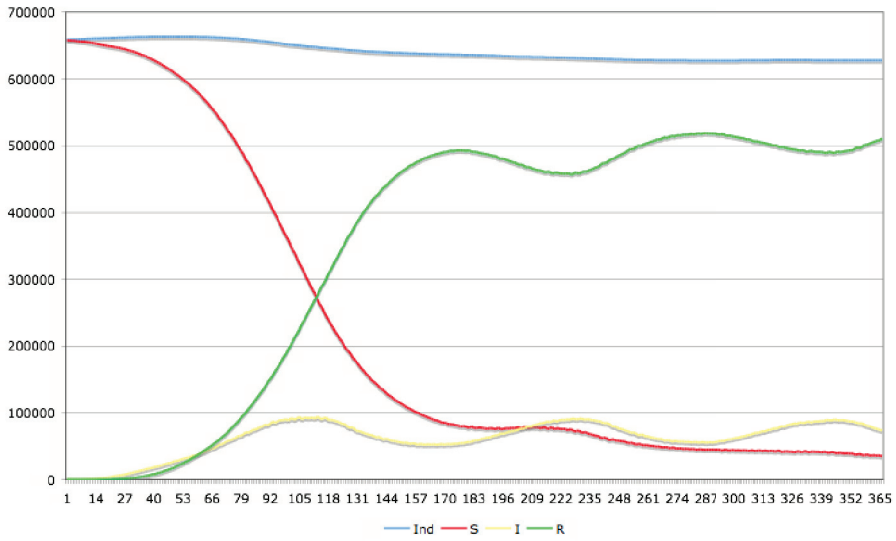


Fig. 9. Scenario A. Medical treatment is performed, and social behavior changes during the arising situation.

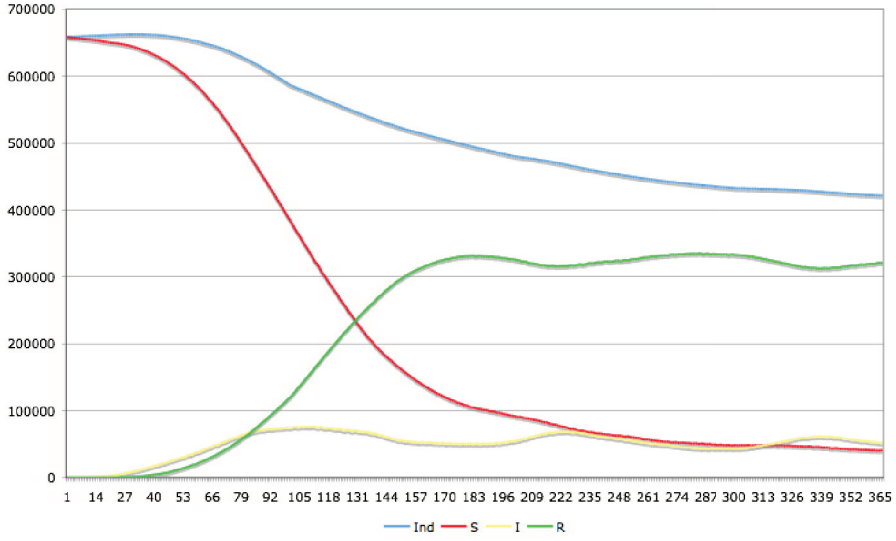


Fig. 10. Scenario B. No medical treatment is performed. Only the social behavior changes during the simulation run.

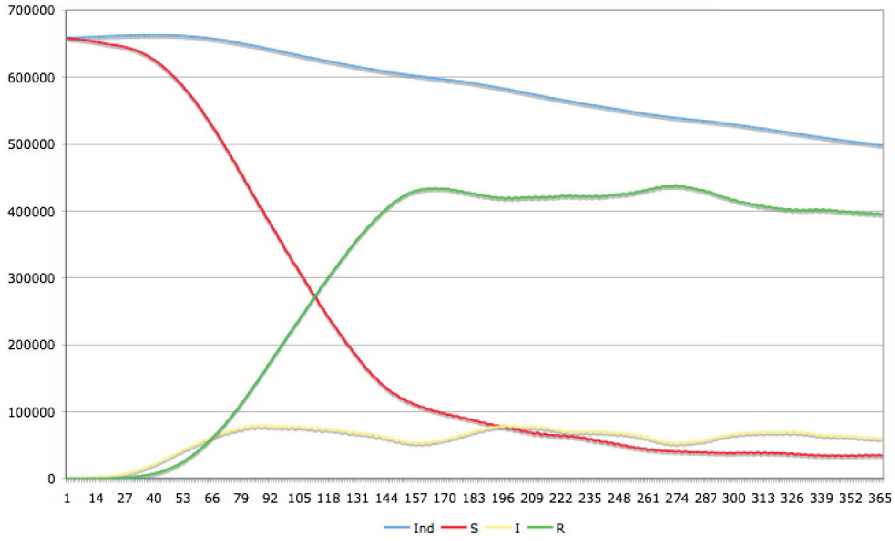


Fig. 11. Scenario C. Medical treatment is performed, but no changes in the individuals' behavior is simulated.

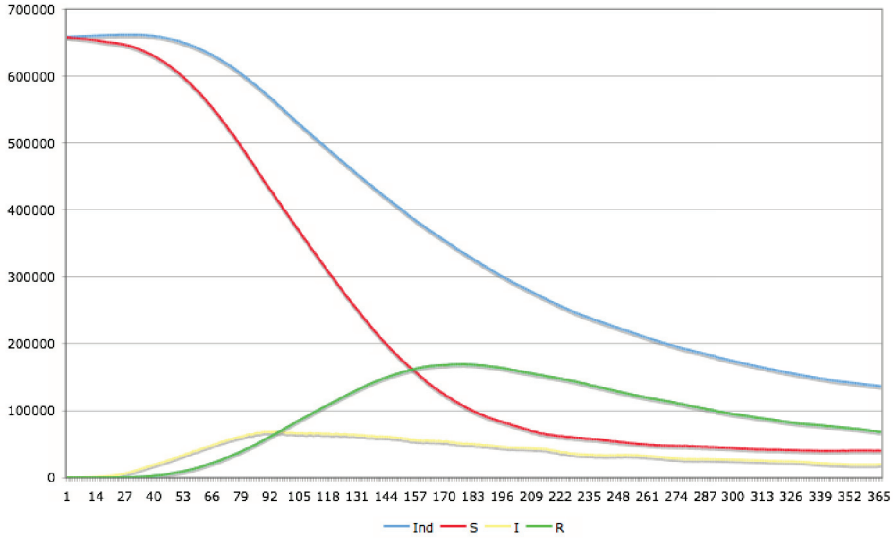


Fig. 12. Scenario D. No medical treatment and no change in the behavior is applied.

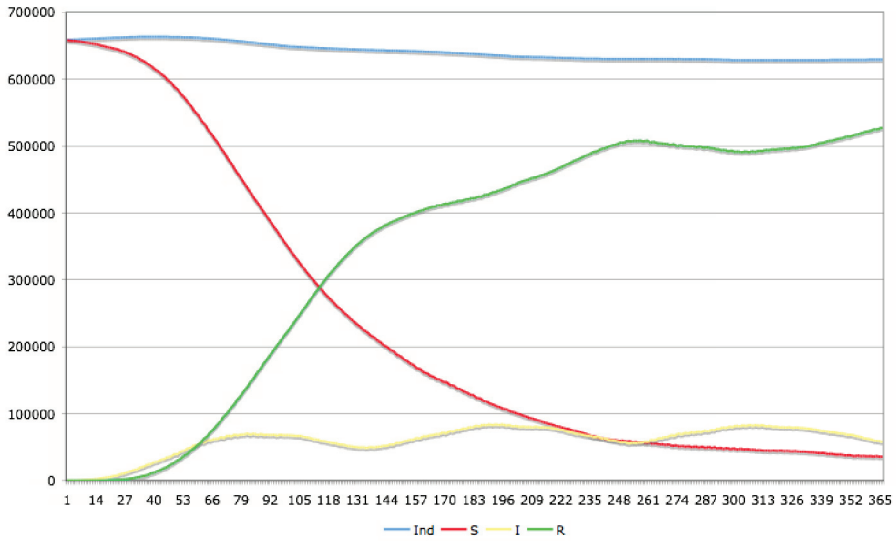


Fig. 13. Scenario E. Equal to scenario A with the difference, that after 50 days a controlled quarantine is applied.

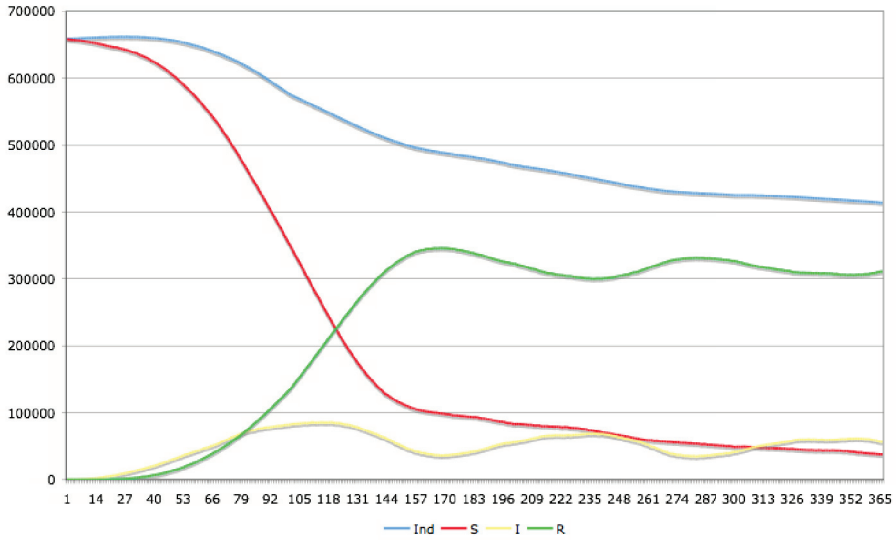


Fig. 14. Scenario F. Equal to scenario B with the difference, that after 50 days a controlled quarantine is applied.

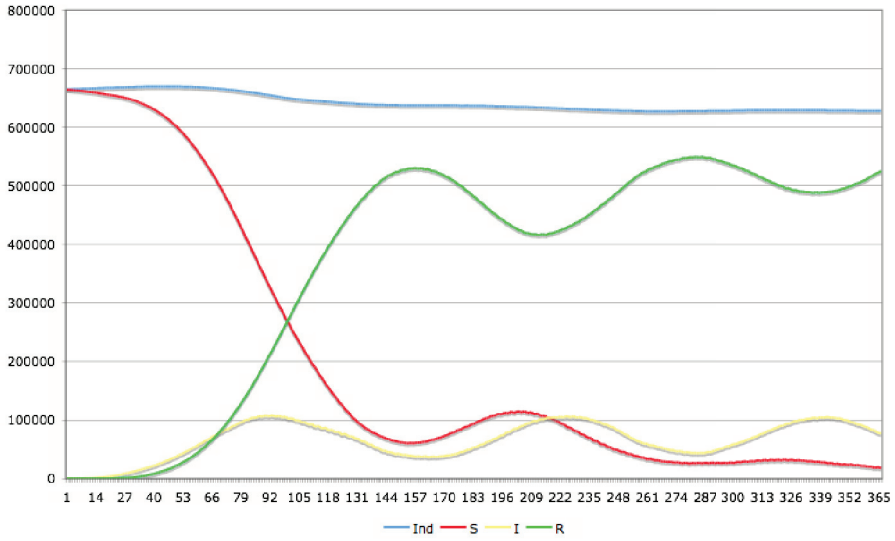


Fig. 15. Scenario An. Equal to scenario A with the difference that no geographical information was used. The population was therefore homogenous.

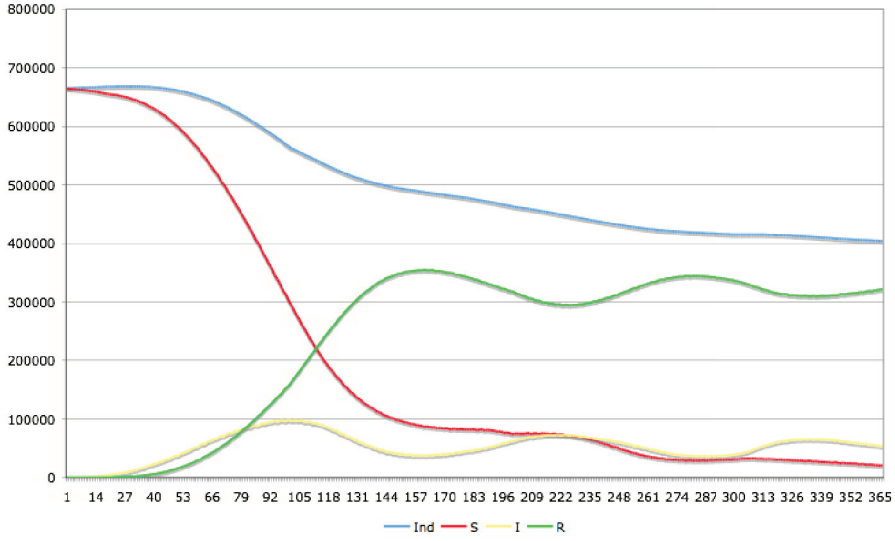


Fig. 16. Scenario Bn. Equal to scenario B with the difference that no geographical information was used. The population was therefore homogenous.

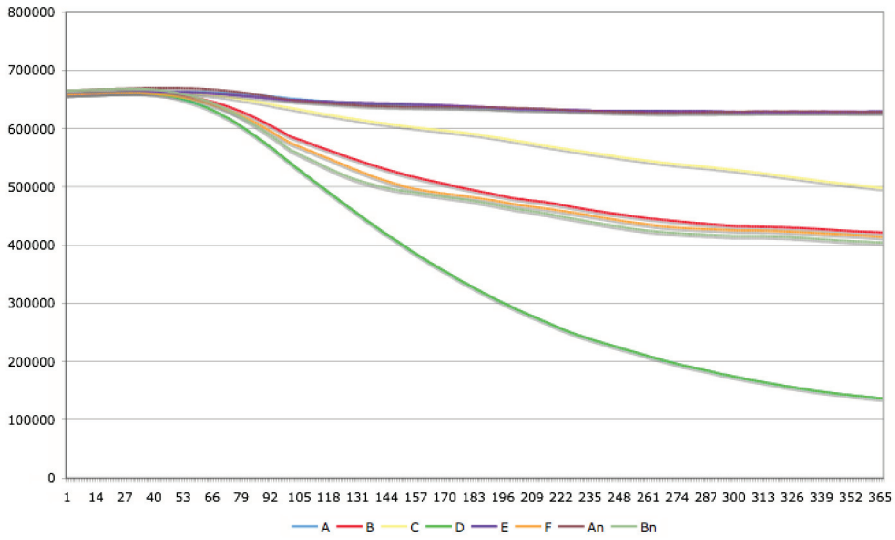


Fig. 17. Population change over the time.

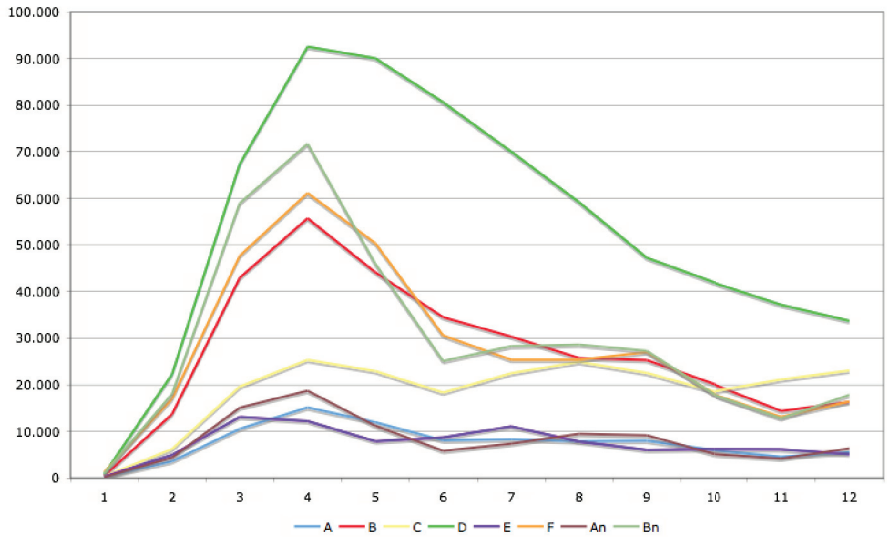


Fig. 18. Fatal cases aggregated per month.

death and diseases fatal cases for the simulated scenarios A to F and An, and Bn. The presented values are mean values per day.

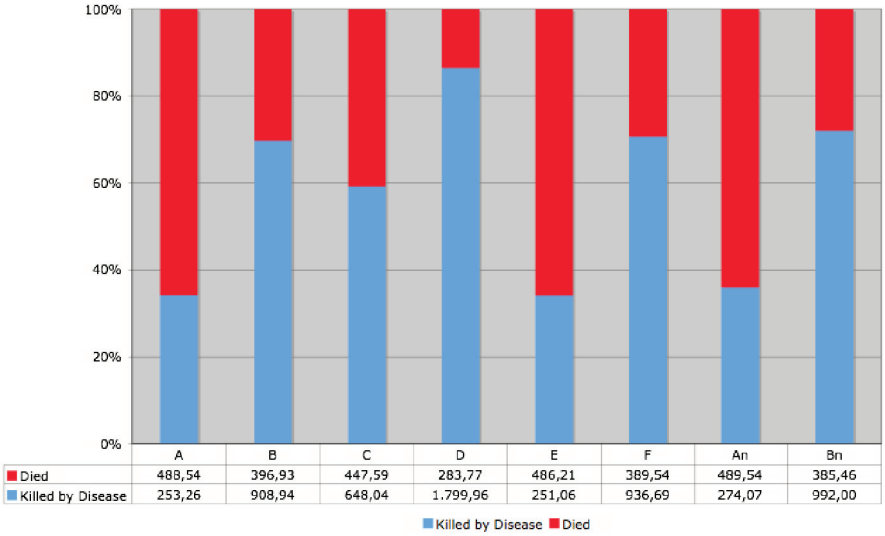
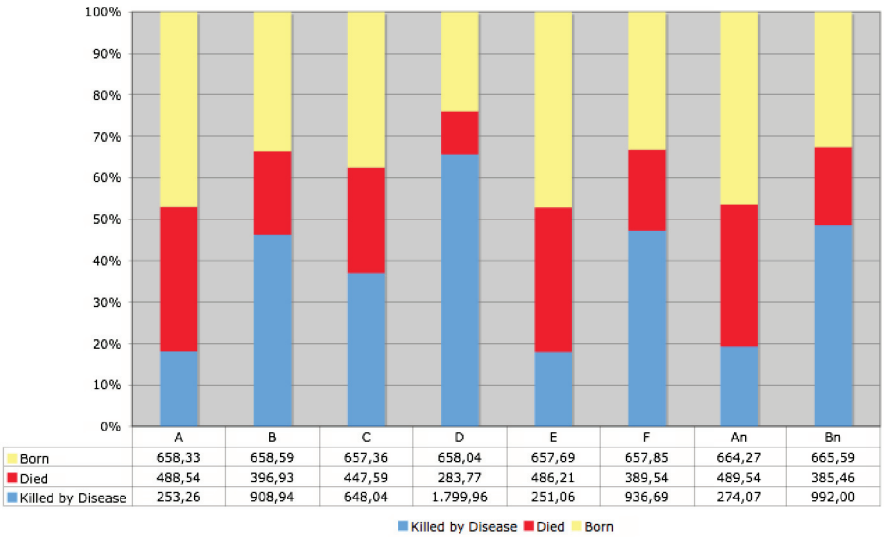


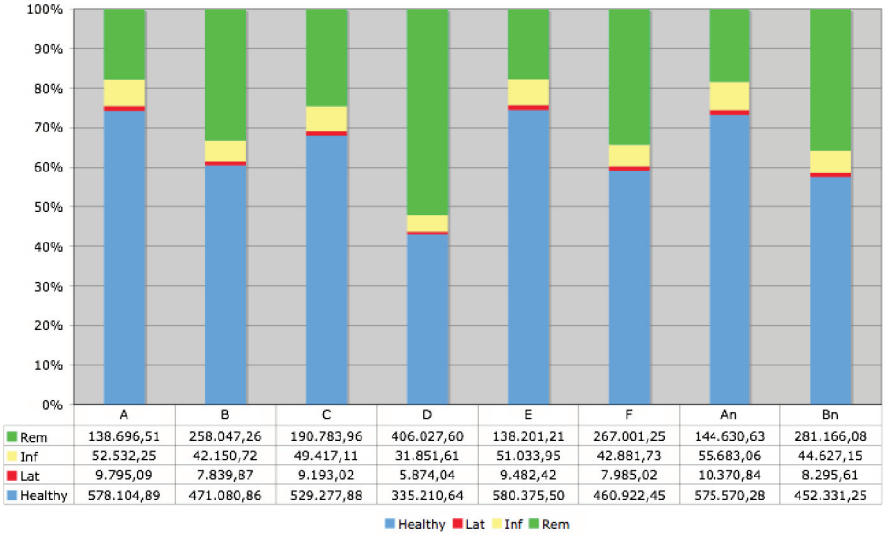
Fig. 19. Percentage diagram with mean cases per day. The figure shows clearly, that in the scenarios B, Bn, F, and especially D the death cases caused by the disease prevail.

When adding the natural births, it is possible to predict the population development during the disease spread. Figure 20 shows, as expected, that the population decreases in each scenario, but scenario D shows a dramatic decrease in the population.
The parameters used for the diseases life cycle were 3 days for the latent period, 10 days for the infectious period, the incubation period was set to 3 days and the symptomatic one to 4 days. After 15 days, the individual get removed or recovers from the disease. Figure 21 shows the ratio between the disease life cycle states. It is clearly visible that in simulation scenario D the most infections occur and the disease is able to spread the most. Furthermore, when comparing the scenarios A with An and B with Bn it can be figured out that the presence of geological and demographic realities reduces the spreading in a natural way.
The pie chart presented in figure 22 shows the perceptual distribution of the fatal cases per day.
The simulation showed that the geographic structure of the area is of importance as natural barriers slow down the velocity of the spread. A slower velocity coupled with the change of the natural behavior of the individuals helps to reduce cases of death. Because in this simulation the temporary immunity was set to 100 days, the figures show a periodicity with decreasing amplitude.
As a cellular automaton has a cellular space or cellular lattice, these models allow the visualization of the automaton states at each time point. The regular lattice consists of several individual cells, which interact using a neighborhood relation. In this simulation, a Moore neighborhood instead of a von Neumann neighborhood or 2-radial neighborhood was taken for the simulation. It has to be noted down that when simulating the scenarios using a



	A	B	C	D	E	F	An	Bn
Born	658,33	658,59	657,36	658,04	657,69	657,85	664,27	665,59
Died	488,54	396,93	447,59	283,77	486,21	389,54	489,54	385,46
Killed by Disease	253,26	908,94	648,04	1.799,96	251,06	936,69	274,07	992,00

Fig. 20. Development of the population during the outbreak within the different scenarios.



	A	B	C	D	E	F	An	Bn
Rem	138.696,51	258.047,26	190.783,96	406.027,60	138.201,21	267.001,25	144.630,63	281.166,08
Inf	52.532,25	42.150,72	49.417,11	31.851,61	51.033,95	42.881,73	55.683,06	44.627,15
Lat	9.795,09	7.839,87	9.193,02	5.874,04	9.482,42	7.985,02	10.370,84	8.295,61
Healthy	578.104,89	471.080,86	529.277,88	335.210,64	580.375,50	460.922,45	575.570,28	452.331,25

Fig. 21. Tracking of the disease life cycle for the individuals for the different simulation scenarios.

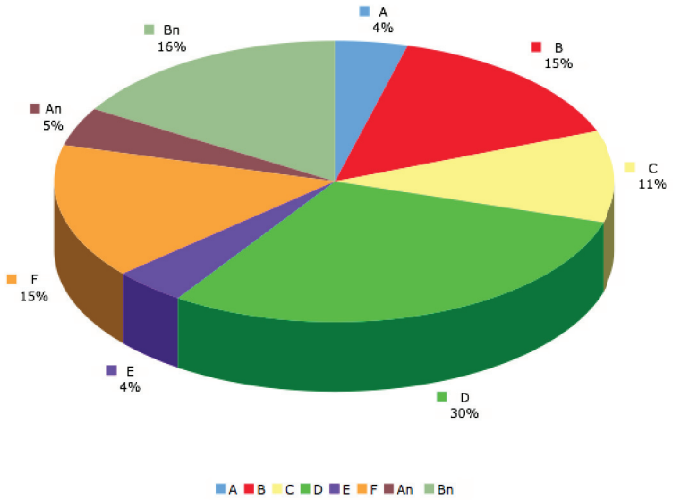


Fig. 22. Percential distribution of fatal cases per day computed by using mean value.

2-radial or von Neumann neighborhood the graphical representation is smoother and ringlike compared to the presented ones, but the overall behavior is almost equal.
Figure 23 depicts the spatial results of the disease spread for scenario A. When analyzing the spreading, one can see that the geographical properties and the different densities in the valleys of Tyrol are responsible for slowing down the outbreak. However, the geographical conditions are unable to stop the outbreak completely.

3.5.3.3 Computing time

The run time of the simulation for the state Tyrol simulation was $\mu = 51.87\ [min]$ minutes ($\sigma = 1.87\ [min]$) per scenario (simulation of 365 time steps per scenario) with four scenarios started parallel on the same machine. The run time of the simulation for Austria was 5.4 $[h]$ hours per scenario. The simulation was performed using an Apple X-Serve 1.1 OS X Server Version 10.4.10 with 2 x 2 GHz Dual Core Intel Xeon processor and 2 GB of RAM installed. The mean storage per scenario was about 10 MB (spatiotemporal snapshots per time step and overall information). The maximum capacity of the framework is only limited by the available memory. A simulation with 10 billion individuals, which is enough to simulate a spread of a disease over the whole globe, would be possible, however, the simulation time would be high. Since the insights, which wanted to be obtained on mechanisms and procedures of disease spread, are based on extensive and complex simulations, it is of great importance to have the possibility to run a large number of simulations or simulations on fine-grained models in a short period without the worry of long simulation periods. It is vital to have the opportunity to experiment with different parameters of the models, in terms of vaccination strategies, behavioral patterns of individuals, model variations and so on. At the European Grid Conference in Amsterdam a framework Wurz & Schuldt (n.d.) for seamless parallel execution of various kinds of algorithms was published. The framework can be used to schedule execution of software in parallel without the burden for the application developer

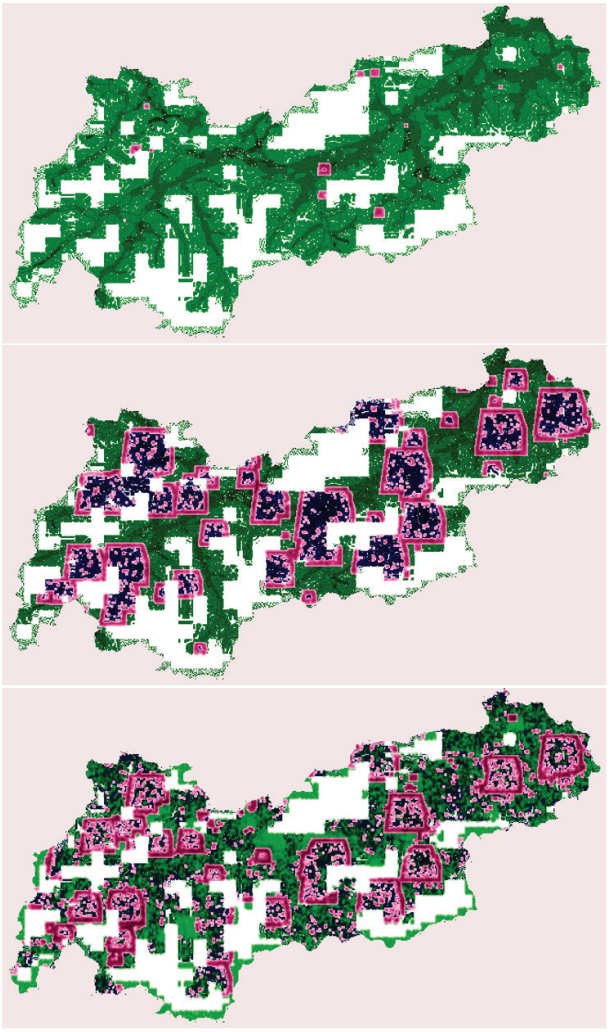


Fig. 23. The colors were used as following: green describes individuals in the state susceptible (S), gentle-pink marks individuals as in state infective (I) and dark blue marks the individuals as to be in state recovered (R). The first graphic depicts the simulation after 20 days. Although the simulation has started in the capital of Tyrol, in Innsbruck, after 20 days there are some outbreaks in the west of the state and in the east, which results form the fact that individuals are moving from cell to cell. Furthermore, unknown activities, which are denoted as spontaneous infection, are responsible for these characteristics. The second image shows a snapshot 80 days later at the time point 100 days after the infection started. It is clearly visible that in the capital where the disease spread started most people are in state recovered. Individual who did not die from the disease do have a temporary immunity. More than 50% of the individuals are in state infected. The last image shows a snapshot at time point 200 days after outbreak where individuals may get infected again. This represents the second outbreak wave with smaller amplitude.

to know details about the computational resources available, and that the framework, acting as a middleware, allows for a dynamic adaptation of the scheduling process.The framework is designed to cope with heterogeneous resources in a dynamic and rather instable network, so that it can be used to utilize computation power available on the Internet or the universities network to speed up simulations of the CA Framework or add additional facets like visualization Ôon the flyÕ. The usage of grid computing in CA model simulation lends itself to be perfect as CA models can be parallelized perfectly. In the first simulation, an Apple Xgrid environment was used. Apple's Xgrid technology enables to use ad hoc groups of Macintosh system into low-cost supercomputer.

3.6 Summary

The framework enables the simulation of different communicable diseases by specifying the disease parameters and demographic characteristics. Furthermore, the population can be divided into subgroups, which enables to simulated different impacts of the disease on each individual. The described scenarios demonstrated that CA and agent based models can be used for simulating and visualizing the spread and the adherent impact of infectious diseases. Furthermore, the simulation environment allows the access to any individual parameter at any point in time of the simulation, which enables detailed statistical analysis. Although the connections and the affiliated behavior between the individuals can be modeled using different neighborhood relations, the behavior of any individual in these models depends on functions using random numbers. For upgrading the behavior algorithms, social behavior approaches should be used and the computation of the economic impact should be computed for better creation of public health strategy plans for managing fatal diseases. The natural manner Ð let us call it "Groundhog Day" - that most individuals do have a way of living caused by their daily workflow, can not be modeled correctly, using such functions. To solve this problem, virtual worlds could be helping. The well-known Second Life, where millions of people do live an additional life and do also have a behavior, which is very similar compared to their own, should be observed for simulations. One has to keep in mind, that building a population model from census and demographic data statistically equal to one in the real world would be very complex but also a deep impact to privacy, and therefore, afflicted with many of problems.

Recapitulatory one can state, that the proposed CA framework is able to support public health offices by providing them with information for creating plans to manage such situations and to prevent serious long-term economic repercussions.

4. Imaging of the cardiac electrical function using cellular automaton approach

4.1 Introduction

Simulation of the electrocardiogram (ECG) and of the body surface potential (BSP) have been a research topic during the last decades. Nowadays, because of the enormous computer power available and because of extensive knowledge about cardiac electrophysiology from the cellular to the tissue level, sophisticated three-dimensional approaches have been developed. Although, the models became very attractive during the last years, there are still extensive problems in making them useful for cardiovascular diagnosis and therapy. First, the individual parameters like fibre architecture, conductivities and others are not available to that extension needed. Second, today, the used cell membrane models have to consider molecular function as well. Finally, the models should be validated based on human data from the cellular to the organ level. This will imply further research necessary in the upcoming two

decades. The paper presented deals with an anisotropic ventricular and an isotropic atrial model developed by our research group during the last 15 years. This in silico approach is used in the electrocardiographic forward and inverse approach. Validation has been done for the inverse formulation in 45 patients only. The whole-heart model presented uses the bidomain source-field formulation. A detailed anatomical model of the atrium is considered as well. The geometrical data is derived from individual magnetic resonance images. Fibre architecture in the ventricle and anatomical features in the atrium, like Bachmann bundle or others are considered based on literature data. The whole-heart model allows the calculation of the de- and repolarization and of the three-dimensional potential pattern throughout the entire heart muscle and volume conductor. Based on this simulated potential data, the 12-lead standard ECG or different BSP maps can be visualized. Today, this in silico whole-heart model environment is used for enhancing the understanding of the nature of the ECG in the normal beat, for different arrhythmias, and for ischemia and infarction. A user-friendly software environment allows interactive model generation, parameter adjustment, simulation and visualization.

4.2 Methods: the forward problem

4.2.1 Volume conductor model

The volume conductor model (VCM) with the embedded cardiac source volume is the basis for the electrocardiographic forward problem. The VCM consists of the compartments chest, lungs, atrial and ventricular myocardium, and of the blood masses from an individual patient. The morphological imaging data were acquired using a Magnetom Vision Plus 1.5 Tesla scanner, Siemens Medical Solutions, Erlangen, Germany. For the lung and torso shape extraction a T1 flash, non-contrasted axial data set during breath-hold (expiration, 10 mm spacing) was used. The cardiac geometry (atrial and ventricular models) was acquired in ECG-gated cine mode during breath-hold (expiration, oblique short-axis scans) with 4 and 6 mm spacing. The segmentation of the compartments was performed using a recently developed VCM segmentation pipeline. The resulting labelsets were triangulated using a standard marching cubes algorithm and optimized using Hammer B. (2001). In the next VCM assembling step the tetrahedral mesh was created using the software package Hypermesh (Altair Eng.), which allows to produce optimized tetrahedral meshes on the basis of the high quality surface mesh. The whole heart VCM consists of 104,001 tetrahedrons and 18,170 nodes, respectively. The volume conductor (whole heart) model with its compartments is depicted in Fig. 24.

4.2.2 Computation of cardiac activation sequences - the cellular automaton

The CA used in this study was developed Fuchsberger M. (1993); Killmann R. (1987; 1990); Rosian M. (1991), modified and implemented in amiraDev 3.0™ Hayn D. (2002). Briefly, after segmentation, triangulation and tetrahedral mesh generation different types of tissue were assigned to the atria and ventricles with corresponding parameter setting for each tissue type. Further, the CA needs the fiber structure assigned to each node of the tetrahedral model as well as the refractory periods assigned to each tissue type.

In the models the types of tissue were the endocardia, epicardium and myocardium. Furthermore, the sinus node, crista terminalis, Bachmann bundle, fossa ovalis, pectinate muscles, coronary sinus, isthmus, the bundle of His, left and right bundle branch and the Purkinje fibers were defined. Each type of tissue has its own set of parameters needed for the computation of the activation times and the time dependent transmembrane potential distribution. The fiber geometry was chosen such, that it fitted qualitatively well with the

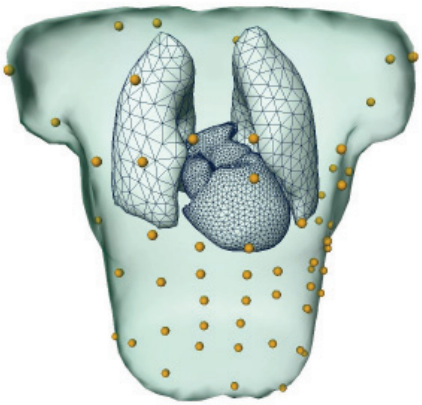


Fig. 24. VCM of an individual patient comprising chest surface, atria and ventricles with the cavitary blood masses and both lungs displayed in an anterior-posterior view.

Fig. 25. Fiber orientation assigned to each node of the tetrahedral ventricular model for four different views. The atria and ventricles are displayed in a transparent style.

findings described in literature Greenbaum R. A. et al. (1981); Mc Veigh E. et al. (2001); Nielsen P. M. F. et al. (1991); Rijcken J. et al. (1999); Streeter Jr. D. D. et al. (1969). Figure 25 gives an impression of the ventricular fiber structure used in this study.
Another essential input for the CA are the effective refractory periods ERP defined for a heart cycle length (CL) of 700 ms (termed ERP_{700}). These can be adjusted for each tetrahedron individually or for tetrahedrons belonging to one type of tissue. The effective refractory period is defined as the time during which the cell cannot be excited by a stimulus of such magnitude, which is twice as high as a stimulus (diastolic threshold) capable to excite a cell in

resting status. The ERP is computed during run time, as the values depend on each preceding diastolic interval and on the current CL (detailed explanation can be found in Killmann R. (1990); Wach P. et al. (1989)). It should be stated that the CL in this context is the time distance between two proceeding excitation processes in one tetrahedron and does not necessarily refer to a CL provoked by any pacemaker (e. g., sinus node, atrioventricular node or paced rhythms). The run time ERP at the CL is calculated by

$$ERP(CL) = ERP_{700} + A + B \cdot \frac{CL - 700}{1000} - D \cdot \left(\frac{C - CL}{C} \right)^2$$

if $CL < C$ and

$$ERP(CL) = ERP_{700} + A + B \cdot \frac{CL - 700}{1000} \tag{10}$$

if $CL \geq C$, respectively. Parameters A (offset value), B (slope of the function), C (a predefined CL), and D (coefficient for the quadratical decrease) have to be defined for each type of tissue individually. The values for these parameters were chosen for each tissue type according to Killmann R. (1990). ERP_{700} was set to 210 ms for the bundle of His, 290 ms for the conjunction points, 280 ms for the left and right bundle branches, 320 ms for the Purkinje fiber network and 260 ms for the ventricular myocardium. Parameter A is introduced for enabling a simple change from physiological to pathological conditions for each tissue type individually. The value for the slope parameter B was obtained by averaging data reported for human hearts. Parameters C and D were set unequal zero for the ventricular epi-, endo- and myocardium only as the relationship between CL and refractory periods shows up to have a quadratically decreasing shape for these ventricular tissue types Killmann R. (1990) for the case $CL < C$. If $CL \geq C$, the $ERP(CL)$ is assumed to increase linearly with the CL.

Computation of activation times

After selecting an arbitrary number of tetrahedrons assigned with an arbitrary time instant for starting the excitation process the activation sequence is computed as follows. Every tetrahedron of the cardiac model can take up three different states: *excitable* (***e***) , *refractory* (***R***; i. e., excited) or *waiting* (***W***; i. e., awaiting excitation). The last state has no physiological meaning but was introduced due to algorithmic needs.

The simulation starts by selecting that tetrahedron with earliest excitation time and its status is changed from ***e*** to ***R***, i. e., the number of this tetrahedron is written into table ***R***. The 'possible excitation times' of the excited tetrahedron's neighbors are calculated according to the distance between the center of masses and conduction velocities for each connection and stored in table ***W***. In case no conduction between two neighboring tetrahedrons can occur the conduction velocity is set to zero. Whether a conduction between the tissues the tetrahedra are belonging to can or cannot occur is stored in an additional parameter set. In the next step the two tables ***S*** and ***W*** are searched for the tetrahedron with the lowest excitation time. This is the next tetrahedron to be excited and stored in table ***R*** and the corresponding table ***W*** is cleared. If two or more tetrahedrons have the same starting or possible excitation time one of them is arbitrarily chosen and the other one becomes/the other ones become the source tetrahedron of the subsequent calculations. The propagation is then calculated as described above, but now three cases may occur: The neighboring tetrahedron is

- in status ***e*** $\rightarrow$ the possible excitation time is stored in table ***W***, i. e., the tetrahedrons's status is set to 'awaiting excitation',
- already in waiting status and has been assigned a possible excitation time with

  - a lower value than the one calculated this time → no changes occur,
  - a higher value than the one calculated this time → the possible excitation time is changed to the lower value,
- in status $\boldsymbol{R}$ → no value is then stored for this point in table $\boldsymbol{W}$.

Then the tables $\boldsymbol{W}$ and $\boldsymbol{S}$ are searched again and the sequence of instructions is performed for the next source tetrahedron. The duration of status $\boldsymbol{R}$ – the period of time a tetrahedron cannot be set into status $\boldsymbol{W}$ or $\boldsymbol{e}$ – is evaluated employing (10) each time a possible excitation is to be assigned to that tetrahedron. The computation is finished, when no more starting tetrahedrons in table $\boldsymbol{S}$ are left and $\boldsymbol{W}$ is empty or when the excitation time of the current source tetrahedron is higher than the chosen upper limit for simulation duration predefined by the user.

Computation of transmembrane potentials

For modeling different shapes of ventricular and/or atrial action potentials the extended Wohlfahrt formula described in Rosian M. (1991) is used:

$$\varphi_m(t-\tau) = \alpha(t-\tau) \cdot \beta(t-\tau) \cdot \gamma(t-\tau) + K_{10} \quad [\text{mV}]. \tag{11}$$

Parameter t is the current simulation time interval, τ represents the computed activation time, $\alpha(t-\tau)$ [–] describes the shape of the depolarization, $\beta(t-\tau)$ [mV] the phase from the beginning repolarization (enables modeling of a notch right after onset of depolarization) to the plateau shape, and $\gamma(t-\tau)$ [–] characterizes the repolarization process. The value is 0 for all parameters of (11), when $(t-\tau) < 0$. Time $t = 0$ is considered as onset of depolarization (time instant, when half of the depolarization amplitude is reached). With the extended Wohlfahrt formula (11) the time dependent transmembrane potential shape for each node of a cardiac tetrahedron can be calculated based on parameters specifically adjusted for the types of cardiac tissue and based on the 'history' of the preceding activation sequence (relaxing phase, diastolic interval) and the computed activation times.

4.2.3 Extracellular potential computation

Based on the quasi static approximation of Maxwell's equations for electromagnetic field calculation and employing the bidomain theory Geselowitz D. B. & Miller 3rd W. T. (1983) (therefore the subheading extracellular potential computation), the resulting differential equations to be solved are

$$div\left[\boldsymbol{\kappa}_b \, grad\left(\varphi\right)\right] = -div\left[\tilde{\boldsymbol{\sigma}}_{in} \, grad\left(\varphi_m\right)\right] \tag{12}$$

for the cardiac region and

$$div\left[\boldsymbol{\kappa}_c \, grad\left(\varphi\right)\right] = 0 \tag{13}$$

for all other compartments of the VCM. The tensor $\boldsymbol{\kappa}_b$ is the bulk conductivity, i. e., the sum of the electrical effective extracellular and effective intracellular conductivity $\tilde{\boldsymbol{\sigma}}_{in}$. The potential φ describes the extracellular, φ_m the transmembrane potential. The tensor $\boldsymbol{\kappa}_c$ holds the electrical conductivities for all other compartments c. For the ventricular electrical anisotropic conductivities, for the other compartments isotropic conductivities were assumed Bradley et al. (2000).

Applying the FEM, considering the volume conductor model and the related boundary conditions Seger M. et al. (2005) the equations (12) and (13) form together a system of algebraic equations

$$\boldsymbol{R}\boldsymbol{\phi} = \boldsymbol{S}\boldsymbol{\phi}_m. \tag{14}$$

The matrices $\boldsymbol{R}$ and $\boldsymbol{S}$ are the so-called *stiffness matrices*, the matrix ϕ describes the potentials in all nodes of the tetrahedral mesh of the volume conductor, the matrix ϕ_m contains the transmembrane potentials in all source nodes of the ventricles. For forward simulation, the matrix ϕ_m is computed by the CA for discrete time steps. For inverting $\boldsymbol{R}$ on the left hand side of equation (14), the matrix $\boldsymbol{R}$ has to be modified as this matrix is positive *semi*-definite due to the *Neuman's boundary condition* on the torso surface Fischer G. et al. (2002). Therefore, the *Wilson central terminal* is used to define the reference potential on the torso surface Fischer G. et al. (2002) to reveal a positive definite matrix $\tilde{\boldsymbol{R}}$. The inversion of $\tilde{\boldsymbol{R}}$ is performed by a solver based on the *conjugated gradient* method. This consequently leads to the desired potentials in all nodes of the volume conductor model

$$\phi = \tilde{\boldsymbol{R}}^{-1}\boldsymbol{S}\phi_m. \tag{15}$$

Apart from scaling of matrix $\tilde{\boldsymbol{R}}$ no additional preconditioning was performed.

4.3 Results

The environment for simulating the potential data and for visualizing the ECG or BSP maps is based on amiraDev™ (TGS Europe Inc.), which was extended implementing the described functionality using the plugin concept. Thus, a homogenous and user-friendly simulation toolbox could be implemented.
Figure 26 depicts a normal sinus rhythm, whereas in figure 27 an extra stimulus between the right lower and upper pulmonary vein was simulated.

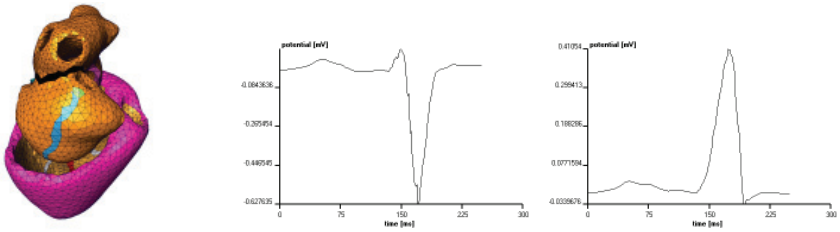


Fig. 26. ECG pattern simulation of a normal sinus rhythm. The gray boxes in the green area (sinus node) mark the tetrahedron where the stimulation starts from. The second figure shows the V3 and the third figure the V5 lead.

By varying the compartment specific parameters (intra- and extracellular conductivities, transmembrane shape) and by specifying the starting point and time any simulation can be performed and used for better understanding the ECG. Furthermore, the potential can be visualized, which also contributes in a deeper understanding of electrical propagation in the heart and the composition of the ECG patterns.

4.4 Discussion

We presented an in silico model environment for the simulation of cardiac de- and repolarization and of the three-dimensional potential pattern throughout the entire volume conductor. A cellular automaton and a bidomain-theory based source-field numerics are the fundamental basics. Only a few simulation scenarios have been presented in this paper. Limitations are given because of the relative course spatial discretization and because of not considering heart muscle contraction. Hence, microscopic cardiac propagation effects can not be simulated. Because we mostly had interest in investigating the macroscopic source-field

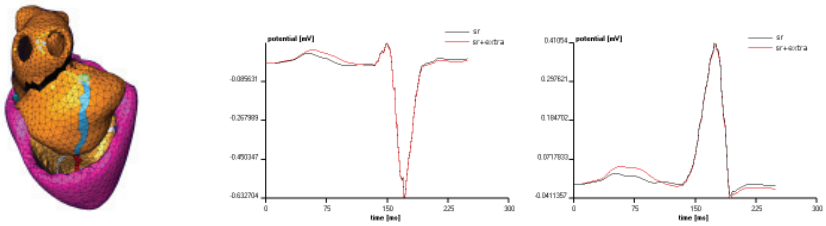


Fig. 27. Simulation of a sinus rhythm with an extra stimulus (staring at 0 ms) between the right lower and upper pulmonary vein. As expected when stimulating the atria at an ectopic focus, the P wave differs compared to the one of the sinus rhythm. This can be seen when comparing the normal ECG (black) with the extra stimulated one (red).

relationship, this limitation can be argued quite well. Because not modelling contraction the simulated T-wave patterns have to be considered as fully synthetic. The simulation of more realistic T-wave patterns have to consider contraction because of the electrical anisotropy and the associated movement of the electrical sources during contraction. Also, we did not consider the very complex fibre architecture in the atrium. Instead of that, for simplicity, we considered electrical isotropy throughout the atrial myocardium. Beside these various limitations, the presented in silico cardiac modelling solution enables various applications for the study of the nature of the ECG pattern in space and time.

5. References

Barrett, C., Eubank, S. & Smith, J. (2005). If Smallpox strikes Portland, *Scientific American* .

Beauchemin, C., Samuel, J. & Tuszynski, J. (2004). A simple cellular automaton model for influenza a viral infections., *J. Theor. Biol.* 232(2): 223–34.

Bossel, H. (1992). Modellbildung und Simulation, *Vieweg Verlag* .

Bradley, C., Pullan, A. & Hunter, P. (2000). Effects of material properties and geometry on electrocardiographic forward simulations, *Annals of Biomedical Engineering* 28(7): p 721–741.

Castiglione, F., Duca, K., Jarrah, A., Laubenbacher, R., Hochberg, D. & Thorley-Lawson, D. (2007). Simulating epstein-barr virus infection with c-immsim., *J Theor Biol* .
URL: *http://bioinformatics.oxfordjournals.org/cgi/content/abstract/btm044v1*

Eubank, S., Guclu, H., Kumar, V. A., Marathe, M. & et al. (2004). Modelling disease outbreaks in realistic urban social networks, *Nature* .
URL: *http://www.mcc.uiuc.edu/nsfitr04Rev/presentations/0113049_Modelling_disease.pdf*

Fischer, G. (2006). In silico cardiac modeling, *Habilitation* .

Fischer G., Tilg B., Wach P., Modre R., Hanser F. & Messnarz B. (2002). On modelling the wilson terminal in the boundary and finite element method, *IEEE Transactions on Biomedical Engineering* 49(3): 217–224.

Fuchsberger M. (1993). *Modellierung der Fasergeometrie in einem numerischen Herzmodell*, Master's thesis, Graz University of Technology. (in German).

Gamma, E. (1994). Design Patterns, *Addison-Wesley Professional - ISBN 0201633612* .

Geselowitz D. B. & Miller 3rd W. T. (1983). A bidomain model for anisotropic cardiac muscle, *Annals of Biomedical Engineering* 11(3–4): 191–206.

Greenbaum R. A., Ho S. Y., Gibson D. G., Becker A. E. & Anderson R. H. (1981). Left ventricular fibre architecture in man, *British Heart Journal* 45(3): 248–263.

Hammer B. (2001). *Optimisation of surface triangulations for image reconstruction from ecg mapping data*, Master's thesis, Graz University of Technology. (in German).

Hayn D. (2002). *Ein finite Elemente Herz Modell*, Master's thesis, Graz University of Technology. (in German).

John von Neumann (1966). The theory of self-reproducing automata, *University of Illinois Press* .

Killmann R. (1987). *Numerisches Herzmodell*, Master's thesis, Graz University of Technology. (in German).

Killmann R. (1990). *Three-dimensional numerical simulation of the excitation and repolarisation process in the entire human heart with special emphasis on reentrant tachycardias*, PhD thesis, Graz University of Technology.

Mc Veigh E., Faris O., Ennis D., Helm P. & Evans F. (2001). Measurement of ventricular wall motion, epicardial electrical mapping and myocardial fibre angles in the same heart, *in* Katila T. & Neonen J. (eds), *Functional Imaging and Modeling of the Heart*, pp. 76–82.

Nielsen P. M. F., LeGrice I. J., Smaill B. H. & Hunter P. J. (1991). Mathematical model of geometry and fibrous structure of the heart, *American Journal of Physiology* 260(4.2): H1365–H1378.

NY (1977). Numerical methods for partial differential equations., *New York: Academic Press* .

Rijcken J., M., B., Schoofs A. J., van Campen D. H. & Arts T. (1999). Optimization of cardiac fiber orientation for homogeneous fiber strain during ejection, *Annals of Biomedical Engineering* 27(3): 289–297.

Rosian M. (1991). *Modellierung der Aktionspotentialformen im menschlichen Herzen*, Master's thesis, Graz University of Technology. (in German).

Seger M., Fischer G., Modre R., Messnarz B., Hanser F. & Tilg B. (2005). Lead field computation for the electrocardiographic inverse problem – finite elements versus boundary elements, *Computer Methods and Programs in Biomedicine* 77(3): 241–252.

Shannon, C. E. (1948). A Mathematical Theory of Communication, *Bell System Technical Journal* vol. 27: 379–423.

Stephen Wolfram (2002). A new kind of science, *B & T* vol. 1.

Streeter Jr. D. D., Spotnitz H. M., Patel D. P., Ross Jr. J. & Sonnenblick E. H. (1969). Fiber orientation in the canine left ventricle during diastole and systole, *Circulation Research* 24(3): 339–347.

Volterra, V. (1926). Fluctuations in the abundance of a species considered mathematically, *Nature* 118: 558–560.

Wach P., Killmann R., Dienstl F. & Eichtinger C. (1989). A computer model of human ventricular myocardium for simulation of ecg, mcg, and activation sequence including reentry rhythms, *Basic Research in Cardiology* 84(4): 404–413.

Wurz, M. & Schuldt, H. (n.d.). Dynamic Parallelization of Grid-Enabled Web Services, *European Grid Conference, Amsterdam* .

Xiao, X., Shao, S.-H. & Chou, K.-C. (2006). A probability cellular automaton model for hepatitis b viral infections., *Biochem. Biophys. Res. Commun.* 342(2): 605–10.

24

Visual Spike Processing based on Cellular Automaton

M. Rivas-Pérez, A. Linares-Barranco and G. Jiménez, A. Civit
Department of Computer Architecture and Technology. University of Seville
Spain

1. Introduction

Cellular organization in biology has been an inspiration in several research fields, such as the description and definition of Cellular Automaton, Neural Networks, Spiking Systems, etc. Cellular Automaton (CA) is a bio-inspired processing model for problem solving, initially proposed by Von Neumann. This approach modularizes the processing by dividing the solution into synchronous cells that change their states at the same time in order to get the solution. The communication between them and the operations performed by each cell are crucial to achieve the correct solution. On the other hand, Spiking Systems (SS) are composed by spiking neurons that receive the information codified into spikes, also called action potential, from several inputs with different weights, and after an internal processing, they produce new spikes in the output that are sent to other spiking neurons. The number and the connectivity between spiking neurons will determine the Spiking System. Weights are programmable and usually learned in order to make effective the result of the network. These weights usually depend on the spike rate of the inputs. The learning process for these weights is called Spike-Timing-Dependent Plasticity. One example of spiking neuron is the Integrate and Fire neuron (Farabet et al., 2009). This neuron can receive weighted spikes from different inputs. It performs internally an addition operation. When the internal value is greater than a configurable threshold, an output spike is produced and the neuron is reset. Researches and engineers try to integrate into a chip several thousand of these IF neurons, but a connectivity problem arises when they try to extract the output of all cells from the chip. The Address-Event-Representation (AER) is a possible solution. AER is a neuro-morphic communication protocol for transferring asynchronous events between VLSI chips that implement a spike-based process typically by using spiking neurons. These neuro-inspired implementations have been used to design sensor chips, like retinas (Lichtsteiner et al., 2008) and cochleas, processing chips (convolutions, filters) and learning chips, which makes it possible to develop complex, multilayer, multichip neuro-morphic systems (Mahowald, 1992).
Both CA and SS have important similarities and they also complement each other. CA is a processing model for problem solving and SS with AER gives a solution for implementing a grid of neurons in hardware. In this chapter our goal is to join both of them and study the viability of this new field for visual processing. Most of the complex visual processing mechanisms are based on solving a set of convolutions for edge detection, contrast adjustment, blur, noise reduction ... But it may also be possible to detect or recognize simple

shapes using large enough convolution filters. In SS the information is codified into spikes, so special sensors must be used or when using conventional digital sensors, special converters must be implemented. A camcorder, for example, offers a frame-based video that consists of a sequence of frames with a normal frame rate of 25-30 frames per second. In contrast a spike-based visual sensor will offer the information codified in spikes, in a continuous way, without the need of frames. These spikes can represent gray levels or any other characteristic, like temporal luminosity changes, contrast changes, etc. There are several published works that demonstrate the
efficiency of spike-based visual processing, e.g. results of the European project CAVIAR, which Spiking System developed was able to sense, detect, filter and learn the trajectory of an object in a totally spike-based way, with a low latency time (<1ms), without frames and performing convolutions for object detection.
Thanks to the CA architecture and its characteristics, it is possible to perform more complex object detection and categorization. Cells in a CA can communicate with their neighbours in a more complex way than IF neurons. Therefore, while a complex visual processing needs several layers of IF neurons in a SS, it is possible to join several layers in a unique CA. A software simulator of spike-based CA is very useful and necessary when trying to implement this processing in hardware.

2. What is Address-Event Representation?

Address-Event Representation (AER) is a spike-based representation technique for communicating asynchronous spikes between layers of neurons in different VLSI neuro-inspired chips, that allows the construction of multilayered, hierarchical, and scalable processing systems. The spikes in AER are carried as addresses of sending or receiving cells on a digital bus. Time represents itself as the asynchronous occurrence of the event. An arbitration circuit ensures that neurons do not access the bus simultaneously. This AER circuit is usually built using self-timed asynchronous logic (Boahen, 1998).
Every time a cell generates a spike, a digital word (address) which identifies the cell, is placed on an external bus. A receiver chip connected to the external bus receives the event and sends a spike to the corresponding cell. In this way, each cell from a sender chip is virtually connected to the respective cell in the receiver chip through a single time division multiplexed bus.

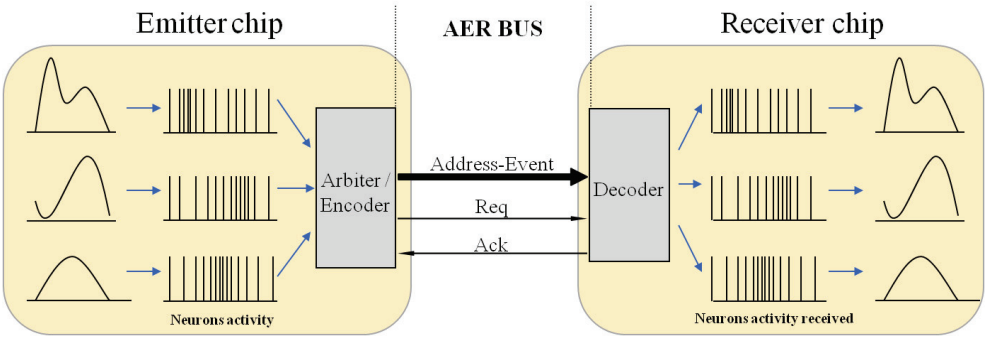


Fig. 1. AER inter-chip communication scheme

The most active cells access the bus more frequently than less active ones, for example, the AER information transmitted by a visual AER sensor is usually coded in gray levels, so the number of events transmitted by a pixel through the bus identifies the gray level of that pixel.

3. Cellular automaton and AER processing

One of the first processing layers in the cortex consists of applying different kinds of convolution filters with different orientations and kernel sizes. These spike-based convolution filters may be developed through an AER system based on Cellular Automaton (AER-CA)

The philosophy of AER systems is lightly different from CA but also similar in a certain sense, so an AER-CA only uses some characteristics of each one. For example, the state of a cell is a function of the current state of the cell and its neighborhood like CA, but a cell only modifies its state when it receives a stimulus.

An AER-CA consists of a 2D grid of cells and every cell is connected to its neighbors. The state of a cell is defined by a set of bits that varies longitudinally when a stimulus arrives. The state of each cell is defined as a function of current state of the cell and its neighbors.

When an AER event arrives, its address is decoded and a spike is sent to the corresponding cell. When a cell receives a spike, its state with an increment of the centre of the KxK kernel coefficient and it sends a spike to its neighbors which increments its state by the corresponding KxK kernel coefficient, i.e. the convolution kernel is copied in the neighborhood of the targeted cell. Therefore, for each spike received, several increment operations are carried out in the neighborhood of the target cell. When a cell reaches its threshold, a spike is produced and the state of the cell is reset.

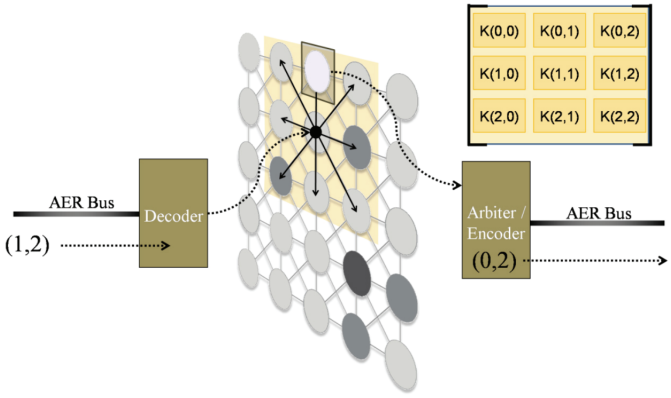


Fig. 2. Behavior of an AER-CA

I4n a Bi-dimensional image convolution, each cell represents a pixel from the image. So, each cell receives a number of events depending on the gray level of the corresponding pixel, so neighborhood increments its state as many times as the gray level and thus the multiplication of the convolution operation is implemented.

An acknowledge signal is sent to the AER emitter when the cell and its neighbors are updated. If the cell state achieves a threshold, such cell generates an event and resets itself.

This behavior corresponds to IF neuron model. Coefficients and threshold are available for all cells.

4. Software simulation of an AER-CA

A first approach to demonstrate the AER-CA behavior consists of using a simulation AER tool. In this section, we describe a self-made simulator to test AER systems whether CA-based or non-CA-based.
AER simulator is an application in C# that offers a friendly interface for simulating AER systems. This application contains a set of basic AER tools which are showed like boxes. These boxes are interconnected and configured to build the AER system to test. Once our design is completed, the simulation starts and results are showed or logged.

4.1 Description of AER simulation software

AER simulator is divided into 2 areas: the control-space and work-space. Control-space shows a set of AER tools that can be used and several buttons to start/stop the simulation, load/save an AER design, etc. Work-space is the area where the AER system design is built.

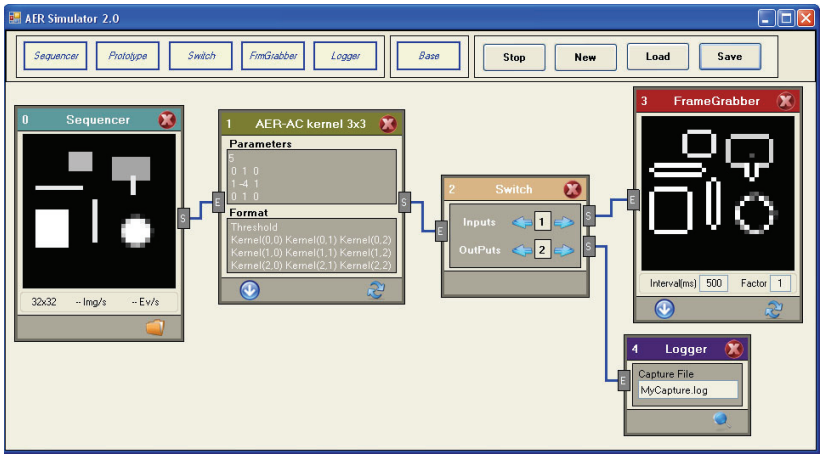


Fig. 3. Snapshot of AER simulation software

During AER system design, AER tools are dragged from control-space to work-space and they are interconnected through their input and output ports to create one or more AER chains. An AER design can contain several tools of the same type.
Every AER tool performs a concrete function in the AER chain. These tools are executed as process that communicate to others process. They can be classified into three categories: sender, actuator (modifier) and receiver device.
Sender devices are tools which generate a new AER event stream and they are used as input to the simulator. AER simulator implements 2 sender devices: Sequencer and DataPlayer. Sequencer generates an AER event stream from a bitmap image that is transmitted cyclically and Dataplayer sends repeatedly an AER event stream specified by a plain text file.
Receiver devices capture received events to be displayed or saved. They are used at the end of the chain to obtain results of the simulation. There are 2 receiver devices implemented in

the simulator: Framegrabber and Datalogger. Framegrabber captures events for a specified integration time and rebuilds an image from these events to be displayed and Datalogger saves received events in a plain text file. Dataplayer can use directly this file.
Actuators are those devices which modify AER event stream received in some way. There are 2 actuator implemented: Switch and Prototype.
Switches can perform 2 different functions simultaneously. They allow distributing received events to several outputs to create new streams from the same source. On the other hand, they can also join several event streams, e.g. they may be used to introduce simulated noise into the design.
Prototypes collect every algorithm to test registered in the simulator and display a list of them. User uses this list to select the prototype to test, so a prototype acts as the selected algorithm from this list by user. This tool also contains a textbox to specify the configuration parameters of the algorithm.
Every algorithm must be implemented in C# and it basically consists of implementing the interface 'Interface Prototype' through a template. Later, this new interface is registered in the simulator to be detected by prototype tools.
The implementation of this interface consists basically of:

a. Establishing in the class the name properties of the device, number of setting parameters, parameters format and example of the format that the user of the application may visualize using the prototype tool.
b. Defining the behavior of the prototype through the implementation of the class method 'Behavior'. The access to input and output ports is carried out using the 'Send' and 'Receive' methods.

Once the interface is implemented, it is added to the simulator in order to be detected by the prototype tool.

4.2 Software simulation of an AER-CA for AER filtering

A first step for the study of the AER-CA is to verify its behavior in the software simulator through the steps described in the previous section.
As it was previously indicated, the prototype behavior is established in the class method 'behavior'. Figure 4 shows the body of this method for a AER-CA for a 3x3 convolution kernel.
Each device of the simulator is executed as a process that is executed indefinitely and communicates and synchronizes with the rest of the processes through the buffers. In the code of figure 4, the prototype receives the events through the input buffer 'MyInBuf' and sends the new events to the next device through the output buffer 'MyOutBuf'. In this algorithm the cell net is represented by a bidimensional matrix, in which each element of the matrix keeps the internal state of a cell. Every time an event arrives, the algorithm executes a double loop in order to modify the state of neighbor cells. If the state of one cell reaches the threshold, they send an event to the exit and this cell resets its state.
In figure 3, window 1 labeled as AER-CA Kernel 3x3 represents the AER-CA prototype implemented for a 3x3 kernel. This prototype uses the release threshold of neurons and the coefficients of the 3x3 convolution matrix as input parameters, as shown in figure 3. The prototype uses a release threshold of 5 and a convolution kernel of [0 1 0 ; 1 -4 1 ; 0 1 0] for edge detection. The device FrameGrabber shows the edge of the image generated by the sequencer after applying the convolution to the image.

```
…
while (true)
{   EvReceived = MyInBuf.Receive();
    for (int i=EvReceived.i-1; i<=EvReceived.i+1; i++)
    {   for (int j=EvReceived.j-1; j<=EvReceived.j+1; j++) // for each i,j
        {   // when i,j are not out of the matrix.
            if ((i>=0 && i<TamMatrix.Y) && (j>=0 && j< TamMatrix.X))
            {   // Adds the corresponding kernel coefficient.
                NewState=State[i,j]+Kernel[i-EvReceived.i+1,j-EvReceived.j+1];
                if (NewState < Threshold)
                    State[i, j] = NewState;
                else
                {   State[i, j] = 0;
                    NewOutEvent.i = i;
                    NewOutEvent.j = j;
                    MyOutBuf.Send(NewOutEvent);
                }
            }
        }
    }
}
…
```

Fig. 4. C# code for simulating the AER-CA

5. Hardware design of an AER-CA

Once the behaviour of the AER-CA is checked in the simulator, we are going to explain a first hardware implementation of the prototype. As seen in figure 5, this hardware will consist basically of an input stage that controls the input AER bus and sends a signal to the corresponding cell of the cellular automaton from the input AER event received. The output stage moderates the signals received from the cells of the cellular automaton and translates them into AER events, which are sent through the output bus. The setting parameters that have been sent by USB are recorded in registries by the USB Controller. The cells of the cellular automaton modify their state from the setting parameters, Kernel Coefficients and Threshold, and the pulses received from the input stage.

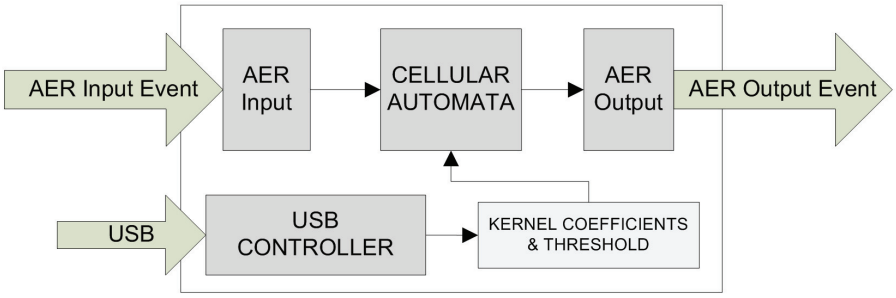


Fig. 5. USB-AER boards and MultiLoadFPGA application

One possible implementation of this system for a 3x3 kernel without the USB Controller is shown in figure 6. AER input is connected to the grid through two-level decoders. When an event comes, e.g. from a vision sensor, its address is divided into row and column. The first level decodes the row by Input Event X and the second level decodes the column by Input Event Y in order to select the targeted cell. Input Req signal notifies when the new event arrived.

In the AER Output, each cell is also connected to a two-level arbiter. The first level consists of a row arbiter and OR gates to encode the output event row address (Output Event X). The second level consists of a column arbiter and an array of multiplexors. This array selects a row from the grid and it is controlled by the first level. The column arbiter encodes output event column address (Output Event Y) from the array of multiplexer. The array of multiplexors and a multiplexor connects the cell to the output. 3x3 kernel elements and threshold are available for all cells of the grid as figure 6 shows. Each cell is connected to its eight neighbors through a single wire, only request signal, to minimize the number of connections.

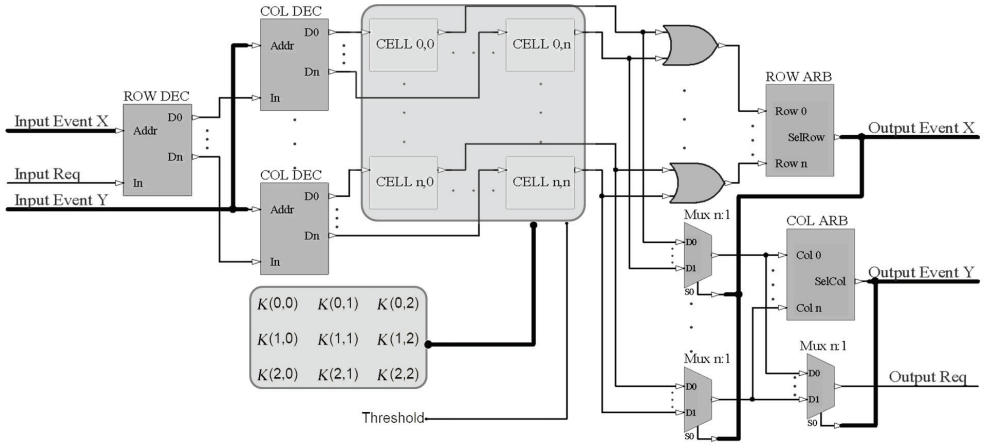


Fig. 6. Block diagram of an AER-CA

Digital logic of each cell in the grid for a 3x3 kernel and 8-wide bus is illustrated in Figure 7. A cell consists basically of two multiplexers to select the kernel coefficient to add (Mux8:1 and Mux2:1), an adder ADD8 to update the cell internal state, a register FD8CE to save the state, a comparator COMP to calculate when the cell must fire, D Flip-Flop FDSR to communicate the cell with the arbiters by handshaking process and some logic gates to control the overflow. The lower overflow is also controlled because some kernel coefficients may be negative.

When a cell receives a spike, it increments its internal state according to kernel center by ADD8 and sends a spike to its eight neighbors simultaneously. Each one of these neighbors adds a kernel coefficient to its internal state depending on where the spike arrives, e.g. when a spike comes from the bottom right cell, it adds the bottom right coefficient of the kernel to its internal state. That is, the cell that receives the spike and its eight neighbors (nine cells in total) modify their states by the corresponding kernel element.

When the state of a cell reaches the threshold, the comparator COMP in the figure 7 resets its state and saves a request in the FDSR. The output arbiter (Figure 6) processes the fired cells by a fixed priority. This arbiter generates the output event address according to the cell attended. When this output event is acknowledged, the arbiter clears the request stored in the FDSR of the cell.

When all fired cells are attended, an acknowledge signal is sent to the emitter AER chip and therefore processing of the incoming event concludes. In this way, no new input events can

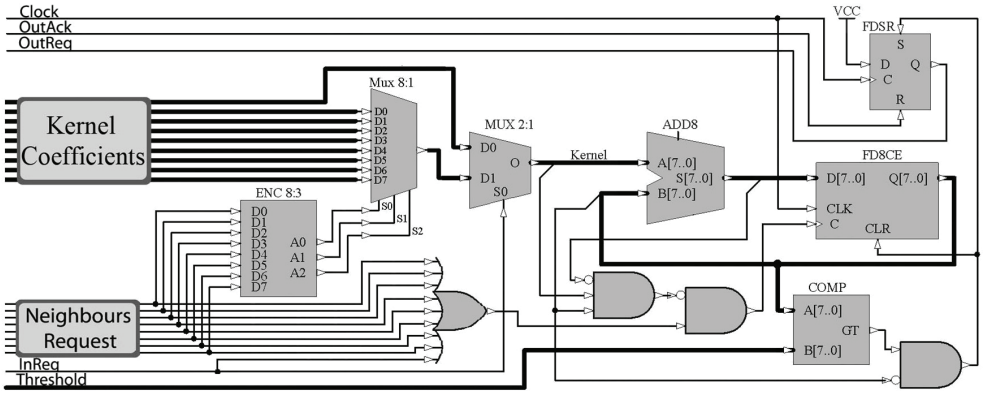


Fig. 7. Block diagram of a cell

arrive before the current event has finished. This constrain simplifies this design because each cell never has more than one pending request so only a Flip-Flop is needed (instead of Input FIFO) to store all new events that arrive while the current event is being processed. Additionally, output arbiter may be fixed priority, which is the simplest implementation, because there is no risk that lower priority cells starve.

5.1 Implementation on a FPGA using glider

The design proposed will be implemented in hardware through a FPGA. The use of FPGAs has spread widely within the last few years both in prototype development research as for commercial implementations thanks to the multiple advantages they involve, such as resetting ease, shorter development time, they provide cheaper and more flexible solutions than those provided by application-specific integrated circuits (ASIC).

The hardware description languages (HDL) were first developed in order to describe the behavior of ASICs and they are used these days to describe the hardware design in FPGAs. For the design of the AER-CA in the FPGA we will use the VHDL language, which is, along with Verilog, one of the most spread hardware description languages.

The description of a cellular automaton through a description language may be difficult because it requires the definition of a great number of cell connections, especially in big cellular automatons. Glider, a graphic application for the description of cellular automatons in VHDL developed by the department of Electronic Engineering of the Technical University of Valencia, will be used in order to simplify the process

5.2 Glider

Glider is a graphic interface that allows: (a) describing a cellular automaton composed of cells and a cellular network, (b) translating the graphical definition into a Cellular Specification Description Language (CSDL) and (c) translating CSDL into VHDL for hardware implementation of the Cellular Automaton (Cerdá et al., 2003 ; Cerdá, 2004).

Glider has been implemented in Java 1.5, which makes it compatible with any platform and operating system.

There are three main descriptions that define Cellular Automaton: cells, cellular network and resource layer. Figure 8 shows the Glider interface for the description of a Cellular Automaton.

Cell: The cell definition consists of defining the set of inputs and outputs, that could come from or go to other cells or could be common for all cells; and defining the operations that cell has to implement with the inputs to generate the outputs. As the cellular automaton depends on the time for the state definition, both sequential and combinatorial logic must be used for a cell. A Cellular Automaton could be formed by several cell definitions.

Cellular Network: The connectivity of cells, the morphology of the network and type of cells used are the parameters that define the Cellular Network.

Resource Layer: This layer defines how a cellular network is connected to another cellular network. It also defines the global inputs and outputs, and the initialization of cells.

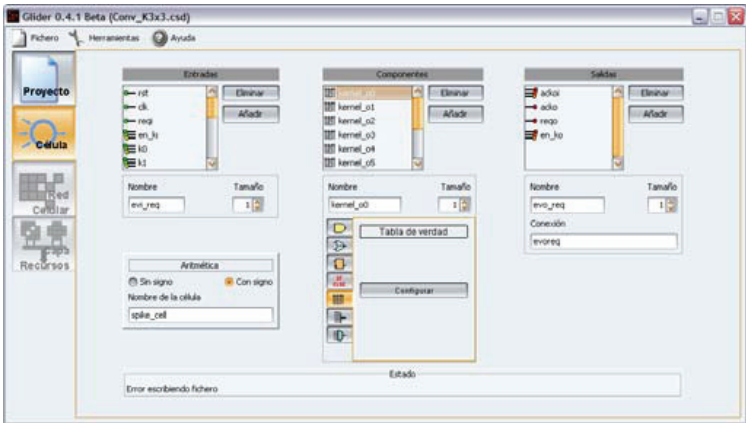


Fig. 8. Snapshot of Glider Java application

Figure 9 shows a block diagram of the Glider internal operation. The graphics interface generates the CSDL files of the Cellular Automaton definition. The CSDL compiler is invoked automatically from this Java application and VHDL files are generated. The console messages are redirected to the application.

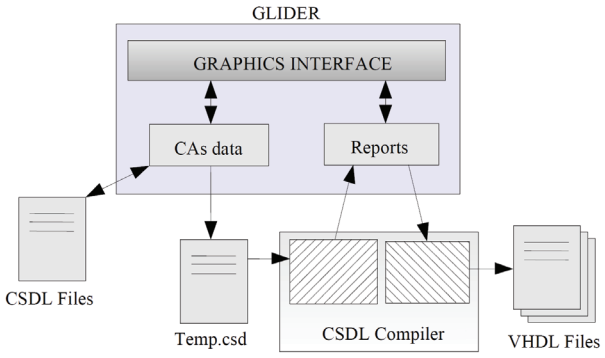


Fig. 9. Internal block diagram of Glider Java application

An interesting capability of Glider is that it keeps continuously checking the consistency of the CSDL generated from the graphics developed by the user. The error or warning messages are reported to the user automatically and in real time.

6. Testing scenario

In order to test this prototype the USB-AER board will be used. This card was developed by the department of Computer Architecture and Technology of the University of Seville for the development of AER devices and it is based on a FPGA Spartan-II 200 of Xilinx.

6.1 AER tools for hardware implementation of an AER-CA

The USB-AER board includes two AER ports (Gomez-Rodriguez et al., 2006), an input AER bus and an output AER bus, connected directly to the FPGA that allows implementing any hardware for manipulating or processing AER information. The Spartan-II 200 FPGA that can be loaded from MMC/SD or USB through the C8051F320 micro-controller. It also contains a large SRAM bank (512Kx32 12ns).

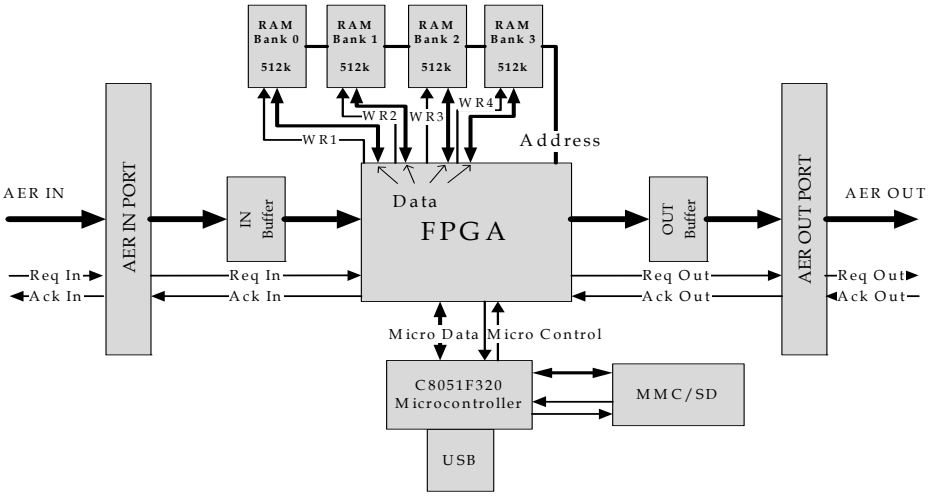


Fig. 10a. USB-AER Board block diagram

The USB-AER functionality depends on the module that is loaded in the FPGA. For example, it may act as a sequencer, monitor, mapping, event processor, data-logger, etc. Most of such functionalities may be performed in a standalone manner. This standalone operating mode requires loading the FPGA and RAM banks from some type of non-volatile storage, e.g. MMC/SD cards. USB input is also provided for development stages.

6.2 Testing AER-CA

Three USB-AER boards have been connected in line, as figure 10b shows, in order to test and measure the performance of these processors. Sequencer firmware is loaded in the first board to send a stream of events which depends on the image previously loaded. The convolution processor under test is loaded in the second USB-AER board. This board

receives spikes from the sequencer and carries out the filtering operation according to 3x3 configurable kernel loaded. The third USB-AER board is loaded with a data-logger firmware to store, in real-time, the time-stamped address of the convolution output events.

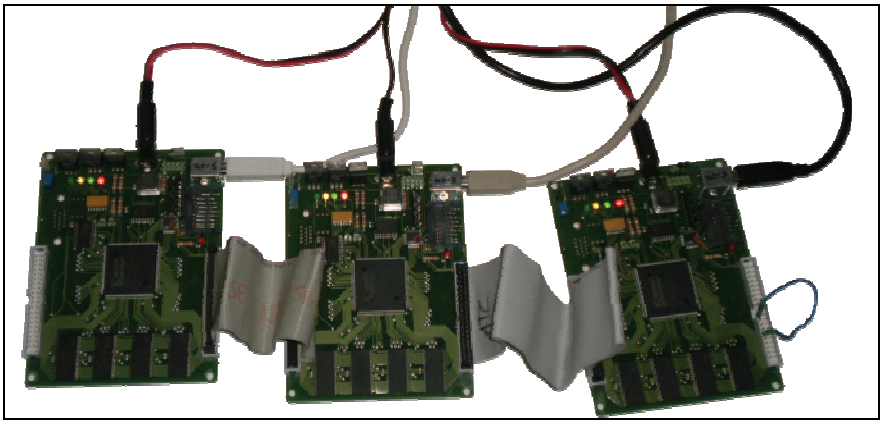


Fig. 10b. USB-AER boards for testing AER-CAs.

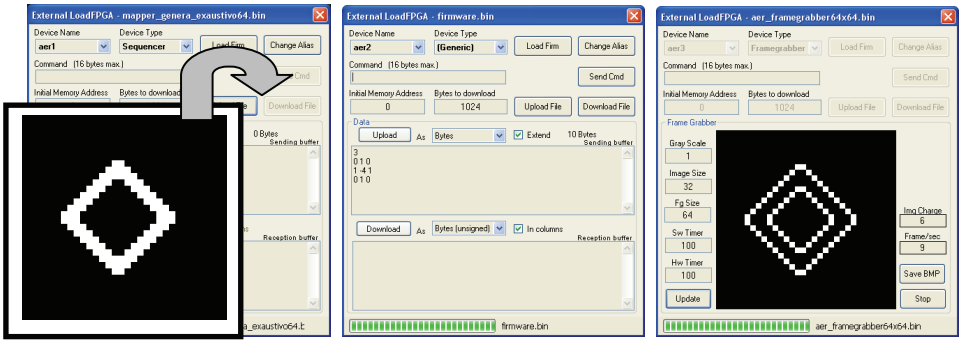


Fig. 11. Snapshot of MultiloadFPGA application

A Windows XP application called MultiloadFPGA (figure 11) controls and configures every USB-AER via USB. Captured spikes can be processed off-line by MATLAB to test the processor and measure performance.

6.3 Performance

The implementation developed requires three clock cycles for every event received: a synchronization cycle, a cycle for edge detection and another one to calculate cell states. Furthermore, this version requires two cycles for every new event generated: one cycle to send an event and another one to wait for the acknowledgement. This implementation yields up to 16.6 mega-events per second when no event is generated. In addition, nine ADD operations are computed every three cycles, thus the system yields up to 150 MOPS when a 50 MHz clock is used.

The event rate achieved is determined not only by convolution processor delay but also by input and output event rate. The input event rate depends on the image loaded into the sequencer and the output event rate is related to the kernel coefficients and threshold.
Figure 12 shows the percentage of the resources used in the Spartan-II 200 of the USB-AER board for different net sizes. Each cell records its state in a 8 bit registry and the kernel coefficients are 4 bit.

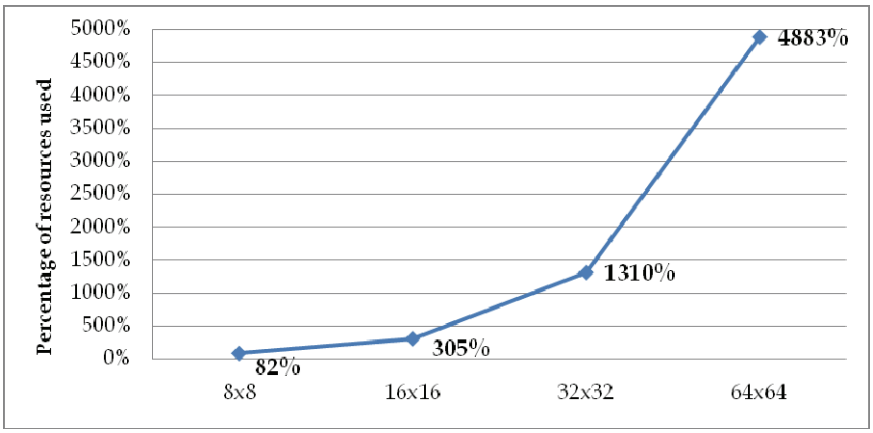


Fig. 12. Percentage of used resources respect to Spartan-II 200

An 8x8 grid spends 84% of all resources in a Spartan-II 200 and it is the largest grid that may be loaded. The resources needed for the AER-CA grow fast with the increase of the net size, e.g. a 32x32 grid requires a FPGA thirteen times larger. This growth is due to the requirement for individual resources of every cell implemented.

7. Resource optimization

The previous implementation requires many resources, so a new implementation is proposed that optimizes resources.

7.1 AER-CA in memory

This new implementation exploits a quirk of the AER-CA: when an input spike arrives, only a subset of cells work simultaneously, the targeted cell and its neighbors. So, it may be possible to maintain the state of every cell in memory and to implement only a shared subset of cells (computational units) that perform the operations needed when a new spike arrives. This new implementation requires less logic but uses memory banks to save the cell states.
Each one of the implemented cells always works with the same kernel coefficient and a different cell but it is the same neighbor regarding to the target cell. Additionally, cell states are strategically distributed to several memory banks so that each implemented cell accesses to a different RAM bank simultaneously.

An example for a 3x3 Kernel and 5x5 grid size is illustrated in figure 13. Each location in the grid indentifies a cell and it is labeled by Bx where x is the number of bank associated to that cell. When the event (2, 3) arrives, the implemented cell (1, 1) uses kernel element K(1,1) and bank B6 to modify the state of cell C(3, 2), the implemented cell C(0,2) uses K(0,2) and B4 to modify the state of neighbor (1,4), and so on until eight neighbors, but each implemented cell accesses to a different bank.
The memory address and memory bank that every implemented cell uses is a function of the corresponding row and column cell. When row and column are divided by 3, quotient corresponds to memory addresses and remainder indicates memory banks, e.g. when the event (2, 3) arrives, neighbor (1, 4) uses B(1 mod 3 , 4 mod 3) = B(1,1) that belongs to B4 (because B(0,0) = B0, B(0,1) = B1, B(0,2) = B2, B(1,0)= B3, B(1,1) = B4, etc).

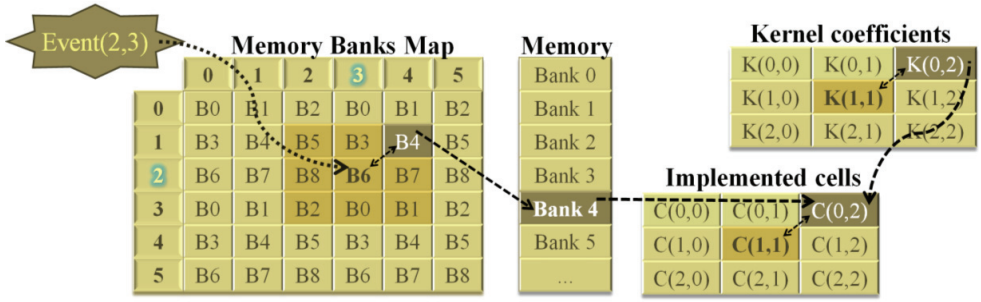


Fig. 13. Example of 3x3 Kernel AER-CA with cell states in memory

7.2 Implementing AER-CA in memory

As it would happen in the first version, the development of element nets in VHDL may be difficult, especially if we want to make several designs with different net sizes, kernel, etc. This is why an application in C# that generates the AER-CA automatically from the desired parameters has been designed. Besides, this application allows to synthesize the VHDL code generated for the FPGA wanted. Figure 14 shows a picture of the application running.
3 areas can be differentiated within this application. The control zone is located at the upper section and it is used to set the design wanted and to initiate the generation and synthesis process. The design parameters that will be used to set the development board are net size, convolution kernel size and the number of bits used to store the state of each cell. The bottom section shows a state window to report the evolution of the process. The left-mid section shows the tree of files generated and the state of the synthesis process. The right section shows the code of the selected file in the file tree. This application also allows to choose the AER-CA implementation desired to be generated, the original version or the memory-based one.

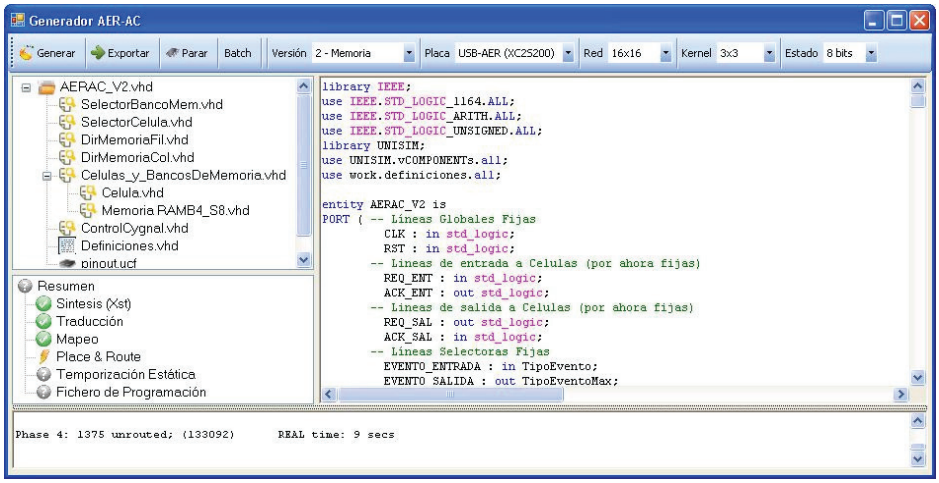


Fig. 14. Snapshot of AER generator application

7.3 Performance

This new implementation saves many resources in comparison to the former one. The diagram in figure 15 shows the percentage of the internal logic of the Spartan-II 200 needed to implement this new version for different sizes of kernel. For greater net sizes the percentage of usage of the Spartan-II 200 increases very slowly. This is because the number of implemented cells does not vary with the size of the net, unlike what occurred in the first version. The number of cells implemented is determined by the size of the kernel.

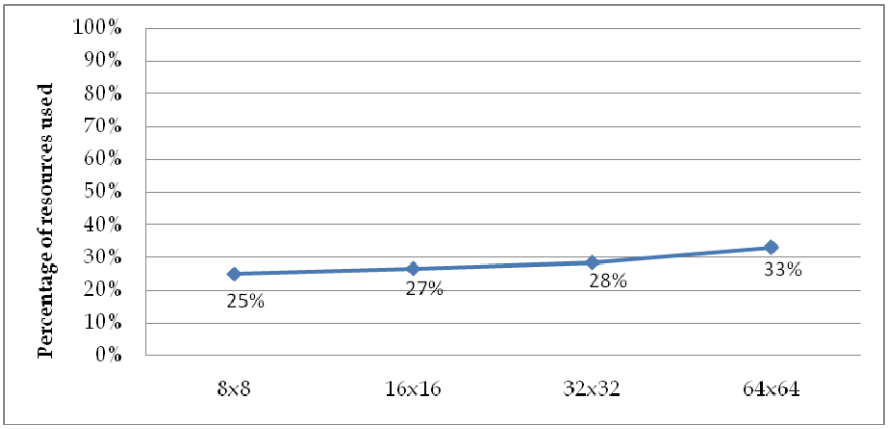


Fig. 15. Percentage of used resources for improved version respect to Spartan-II 200.

In this version, the memory of the FPGA is the one that determines the size of the net since the state of cells is stored in memory. The Spartan-II has 7 Kbytes distributed in 14 memory banks of 512 Bytes. An AER-CA with 3x3 kernel uses 3x3=9 from the 14 memory banks. The biggest net that can be obtained from an AER-CA with 3x3 kernel in a Spartan-II is 64x64

and it uses 4 Kbytes of the 7 Kbytes that the Spartan-II has. AER-CA with Kernel greater than 3x3 requires FPGA with a greater number of banks.
This new implementation requires six clock cycles per received event. Unlike the previous version, this implementation takes four cycles to calculate cell states: A cycle to calculate memory addresses and banks, a cycle to connect each unit to the appropriate bank, another one to read the current state and the last one to write the new state in memory. As the former implementation, this version also requires two cycles per event. It performs up to 8.3 mega-events per second and 75 MOPS.

7.4 Future improvements. A probabilistic AER-CA

Both presented versions require storing the state of every cell. A probabilistic version may be developed to avoid storing states, i.e. a random number and the corresponding probability determine when cells fire instead of adding kernel coefficients until threshold is achieved. This new version may increase processing speed and may be able to work with larger grids in the Spartan-II 200.

8. Conclusion

Cellular Automaton approach has been proposed to develop AER neuro-inspired filters for vision processing. These filters can implement two layers, one for the input processing and the second one thanks to the evolution rule.
AER filters based on 3x3 kernel convolutions have been implemented in VHDL using Cellular Automaton approach.
Two 3x3 kernel convolution AER processor for vision processing inspired in Cellular Automaton have been implemented for FPGA. The original implementation assigns resources to each cell. It performs up to 150 MOPS for a 3x3 kernel and yields up to 16.6 Mega-event per second in a Spartan-II 200. An improved version, that stores the cell states in memory to save resources by reducing number of implemented cells, has also been implemented. This second version achieves up to 75 MOPS and 8.3 Mega-event/s.
A real scenario consisting of three USB-AER tools has been used to prove these implementations and to carry out a performance analysis.
A probabilistic version is suggested to increase processing speed and to reduce resources.

9. References

Boahen, K.A. (1998). Communicating Neuronal Ensembles between Neuromorphic Chips. Neuromorphic Systems. Kluwer Academic Publishers, Boston 1998.

Cerdá, J.; Gadea, R.; Herrero, V.; Sebastià, A. (2003). On the Implementation of a Margolus Neighborhood Cellular Autómata on FPGA. Field-Programmable Logic and Applications. Lecture Notes in ComputerScience 2778, pp. 76-785.

Cerdá, J. (2004). Arquitecturas VLSI de autómatas celulares para modelado físico (in Spanish). PhD Thesis. Universidad Politécnica de Valencia, Valencia, Spain, 2004.

Farabet, C.; Poulet, C.; Han, J.Y.; LeCun, Y. (2009). CNP: An FPGA-based Processor for Convolutional Networks. International Conference on Field Programmable Logic and Applications, 2009. FPL 2009.

Gomez-Rodriguez, F.; Paz, R.; Linares-Barranco, A.; Rivas, M.; Miró, L.; Jimenez, G.; Civit, A. (2006). AER tools for Communications and Debugging. Proceedings of the IEEE ISCAS 2006, Kos, Greece. May 2006.

Lichtsteiner, P.; Posch, C.; Delbruck, T (2008). A 128×128 120 dB 15 μs Latency Asynchronous Temporal Contrast Vision Sensor. IEEE Journal of Solid-State Circuits, IEEE Journal, Vol 43, Issue 2, pp. 566-576, Feb. 2008.

Mahowald, M. (1992). VLSI Analogs of Neuronal Visual Processing: A Synthesis of Form and Function. Ph.D. Thesis. California Institute of Technology, Pasadena, California 1992.

25

Design and Implementation of CAOS: An Implicitly Parallel Language for the High-Performance Simulation of Cellular Automata

Clemens Grelck[1] and Frank Penczek[2]
[1]*University of Amsterdam*
[2]*University of Hertfordshire*
[1]*The Netherlands*
[2]*United Kingdom*

1. Introduction

Cellular automata are a powerful concept for the simulation of complex systems; they have successfully been applied to a wide range of simulation problems Boccara et al. (1993); Canyurt & Hajela (2005); D'Ambrosio et al. (2007); Ermentrout & Edelstein-Keshet (1993); Georgoudas et al. (2007); Guisado et al. (2006); Nagel & Schreckenberg (1992); Popovici & Popovici (2002); Stevens et al. (2007). This work is typically done by scientists who are experts in their field, but generally not experts in programming and computer architecture. Programming complex simulations both correctly and efficiently quickly turns into a painful implementation venture distracting from the far more interesting aspects of the simulation problem itself or the simulated subject matter.

Current advances in computer architecture make the situation even worse. Abundance of parallel processing power through multicore technology requires parallelisation of simulation software in order to effectively use even standard laptop and desktop computing machinery, not to mention clusters of workstations and fully-fledged supercomputing equipment. This situation confronts our dear simulation scientist not only with the task of writing a fairly efficient sequential program, but exposes him or her to the notorious hazards of parallel programming Amarasinghe (2008); Chapman (2007); Gabb et al. (2009). Getting synchronisation and communication requirements correct and deterministic is know to be a far from trivial task, but in fact the problem is even more intricate. Today's hardware reality quickly creates a multi-level granularity problem with different communication and synchronisation abstractions and very different latency/throughput characteristics on each level, from networks interconnecting geographically separated compute centers to multiple cores within the same processor. An experienced programmer can certainly solve all these issues with sufficient time and resources, but the point we make is that scientist we have in mind should not devote his time to this, but rather work on improving his or her model, etc.

The model of cellular automata naturally lends itself to parallel execution following a data parallel approach. This holds not only for multicore processors on the desktop, but likewise for clusters of workstations and parallel computers, in other words on all levels of today's multi-level compute environments. Yet surprisingly little support for programming cellular automata with a focus on high-performace simulation exists. Simulation software is typically

limited in the complexity of cellular automata that can be described: the number of states per cell and the state transition function are typically more oriented towards demonstrations of Conway's game of life Conway (1970) than towards real-life simulations. Furthermore, many approaches seem to focus on the visual aspects of cellular automata rather than simulation performance. Programming complex cellular automata in general-purpose programming languages is not extremely difficult, but it does require substantial programming skills. This hinders the effective utilisation of cellular automata by scientists who are experts in their field, but not necessarily experts in programming.

We propose a new domain-specific programming language named CAOS (Cells, Agents and Observers for Simulation) that is tailor-made for programming simulation software based on the model of cellular automata. Since it is restricted to this single purpose, CAOS provides the scientist with support for the rapid prototyping of complex simulations on a high level of abstraction. Nevertheless, the CAOS compiler fully automatically generates portable and efficiently executable code for a wide range of architectures. We support both shared memory systems through OPENMP Dagum & Menon (1998) and distributed memory systems through MPI Gropp et al. (1994). Both approaches can easily be combined having the compiler generate multithreaded OPENMP code within MPI processes for hybrid architectures. Thus, CAOS not only supports individual multicore processors, but the whole range of computer architecture from laptop processors to supercomputing installations. CAOS allows scientists to harness the potential compute power of both small-scale and large-scale parallel computers for complex simulations with little or even no expertise in parallel programming and computer architecture.

The remainder of this chapter is organised as follows: In Section 2 we introduce the language design of CAOS. Section 4 outlines principles of our implementation while Section 3 provides a brief explanation of the CAOS tool chain. Section 5 discusses a number of runtime performance related experiments. We address related work in Section 6 and conclude in Section 7.

2. CAOS language design

A CAOS program implements all aspects of a cellular automaton simulation. It defines the layout of a multi-dimensional grid of cells, its initialisation (which may also be read from a file) and the behaviour of the cells in the form of a potentially non-trivial state transition function. It also defines how and when snapshots of the simulation are taken and saved. Each of these aspects is implemented in a dedicated CAOS program section. Thus, a CAOS program is organised into a sequence of sections as shown in Fig. 1.

$$\textit{CAOS Program} \quad \Rightarrow \quad \textit{Declarations Grid Cell Init Behaviour} \left[\textit{Observer}\right]^*$$

Fig. 1. General structure of CAOS source files

The *declaration* section contains a number of global decalarations referring to user-defined types, compile time constants and runtime program parameters. Next comes the *grid* section with the definition of the grid, which may have any number of axes, sizes and different boundary conditions. The *cell* section defines the attributes making up each cell's state. The *initialisation* section defines initial values for cells and boundaries. Most importantly, the *behaviour* section defines the state transition function including the definition of neighbourhoods in the cellular automaton. And, last not least, the *observer* section defines how and when snapshots of the cellular automaton are saved to disk while the simulation is running and/or after it has been completed, depending on the concrete application requirements. In the sequel, we will look into each of these sections in greater detail.

2.1 Global declarations

Unlike the other sections the global declaration section itself is a sequence of subsections declaring different objects to be used throughout the remainder of the program. Fig. 2 gives an overview and defines the exact syntax.

Declarations	⇒	⌈*Enum* ⌋* ⌈*Constant* ⌋* ⌈*Param* ⌋* ⌈*Extern* ⌋*
Constant	⇒	**const** *Type Id* = *value* **;**
Param	⇒	**param** ⌈*"String"* ⌋ *Type Id* = *value* **;**
Enum	⇒	**enum** *Id* { *Id* ⌈, *Id* ⌋* } **;**
Extern	⇒	**extern** *Type Id* (⌈*Type* ⌈,*Type* ⌋* ⌋) **;**
Type	⇒	**bool** | **int** | **double**

Fig. 2. CAOS global declaration syntax

The first set of declarations refers to types. CAOS is not a particularly versatile language when it comes to types and type constructors. We deliberately restrict ourselves here; extending the language design by additional machine-supported basic types and standard type constructors like records is merely a matter of engineering, not of language design. In essence, CAOS supports three built-in types: Boolean values (`bool`), machine word integer numbers (`int`) and double precision floating point numbers (`double`). The only type constructor CAOS supports for now are symbolic enumeration types as known from C. We do this to advocate a symbolic style of programming in contrast to a machine-oriented one. Taking Conway's famous Game of Life as a running example, we would suggest to define an enumeration type `dead_or_alive` rather than encoding these different states in Boolean or integer values
The `const` keyword marks the second kind of global declarations; it is used to declare an identifier with a compile time constant value. Again we recommend to define symbolic constants for relevant numerical values to improve the readability of code.
In addition to (compile time) constants CAOS also features simulation *parameters*. These are compile time symbolic values, but runtime constants initialised during program startup. Parameters are declared using the `param` keyword. Like a constant a parameter has a type, a name and a (default) value. The difference is that the given value indeed only is a default value. This default value may be overwritten upon startup of a simulation through a command line parameter. Parameters are a very useful feature if a simulation needs to be run several times for different parameter sets, e.g. from a shell script. Typical applications for parameters are the initialization of

- values of cell components,
- used variables in the cell behavior,
- time steps for observations,

but many others are possible. The keyword `param` may be followed by an optional string enclosed in quotation marks to compile a short description of the parameter into the executable. The information given here will be displayed within the help text that a compiled CAOS program displays if `--help` is passed as command line parameter.
It is possible to use external functions in a CAOS program via the `extern` keyword and a C-style function prototype. The main intended use of this feature is to make mathematical

libraries available to CAOS program to be used in the definition of the state transition function. Fig. 4 shows a some example declarations in a CAOS implementation of Conway's famous Game of Life.

2.2 Grid layout definition

The main characteristic of cellular automaton based simulation is the existance of a multidimensional grid of cells. The grid of cells in CAOS may indeed comprise an arbitrary number of dimensions, each of a potentially different size. The keyword `grid` marks the beginning of this section; Fig. 3 shows the complete syntax.

Grid	⇒	**grid** *Axis* [**,** *Axis*]* **;**
Axis	⇒	*Id* **:** *Size* **:** *Id* **<.>** *Id* **:** *Boundary*
Size	⇒	*IntConstant* | *Id*
Boundary	⇒	**static** | **cyclic**

Fig. 3. CAOS grid layout definition syntax

A grid topology specification is a comma-separated list of axis specifications. Each axis specification consists of four parts separated by colons. Firstly, we have an identifier that names the axis. The second part defines the size of the grid along the given axis. This can either directly be defined by an integer constant or indirectly through a symbolic identifier, thus making use of the constant or parameter mechanism explained before. In particular, the grid size can but need not be fixed at compiler time. Through the parameter mechanism, this vital simulation parameter can easily be supplied at runtime.
The following two identifiers separated by the special symbol `<.>` define names for accessing neighbouring grid cells in direction of decreasing and of increasing indices, respectively. There use will be discussed in detail in Subsection 2.5. The last part defines the boudary condition of the grid, which can either be `static` or `cyclic`. With cyclic boundary condition, the cells on one boundary do have the cells on the other boundary of that axis as their neighbours. With static boundary condition additional constant boundary cells are added to the grid to ensure that all proper cells have a complete neihbourhood. The CAOS example in Fig. 4 continues with a 2-dimensional grid layout declaration where the boundary conditions are cycly, the extent along the x -axis is fixed to 256 cells and the extent along the y-axis defaults to 256 cells, but can be overwritten by a command line parameter at program startup. The directions from any one cell to its neighbours in the grid are named *left* and *right* along the x-axis and *up* and *down* along the y-axis.

```
enum dead_or_alive {dead, alive};
const int x-size = 256;
param "Y grid size" int y-size = 256;

grid:X:x-size:left<.>right:cyclic, Y:y-size:up<.>down:cyclic;

cell {
  dead_or_alive state;
}
```

Fig. 4. CAOS example Game of Life implementation (declaration part)

2.3 Cell attribute definition

After defining the size and topology of the grid, it needs to be populated with cells. The `cell` section defines the attributes of each cell, which can be drawn from the set of built-in types `int`, `double` and `bool` plus previously defined enumeration types. As shown in Fig. 5, syntactically the cell section very much resembles record definitions in imperative languages or the attribute sections of class definitions in object-oriented langauges.
Cell atrributes are readable by other cells from the neighborhood. For example, our running example in Fig. 4 has a single attribute that we simply call *state*, which can be either *dead* or *alive* according to the previous definition of the enumeration type (and the definition of the Game of Life). In general, cells can have a number of different attributes of different types supporting complex state spaces.

Cell ⇒ **cell** { ⌈ *Attribute* ⌋+ }

Attribute ⇒ *Type Id* **;**

Fig. 5. CAOS cell attribute definition syntax

2.4 Grid initialisation

Init ⇒ **init** ⌈*Selector*⌋ { ⌈ *Assignment* ⌋+ }

Selector ⇒ [*Id* ⌈^ *Id*⌋*]

Assignment ⇒ *Id* = *Expr* **;**

Fig. 6. CAOS initialisation section syntax

Before the simulation of the first time step the cells on the grid are initialised. This initial state is defined through a `init` section as shown in Fig. 6 or it may be read in from a file as well. All components of a cell, which are defined in the `cell` section, appear as left hand side of an assignment in the `init` section. Of course, it is possible to leave components uninitialised, but this may obviously lead to erroneous and unpredictable behaviour.
The keyword `init` may be followed by a *selector* to initialize boundary cells at static boundaries. Assume that we would change the grid definition of our example in Fig. 4 to

```
grid:X:1..40:left<.>right:static,Y:1..40:up<.>down:static;
```

Then, to assign values to the cells on the lower boundary of the first dimension, `init[left]` would be used. It is also possible to combine direction specifier. Accordingly, `init[right ^ down]` would initialize the cells that belong to the upper boundary of the first *and* the second dimension.

2.5 Simple state transition functions

In CAOS the state transition function of the cellular automaton is defined by he `behaviour` section. Unlike most cellular automata simulation systems, CAOS provides a fully-fledged, structured imperative programming language to define complex state transition functions. As shown in Fig. 7, we base CAOS on the syntactic foundations of C to faciliate familiarisation.
A simple CAOS behaviour section consists of a sequence of assignments preceded by declarations of local variables. Whereas the declaration of local variables syntactically very

Behaviour ⇒ **behaviour** { ⌈*LocalVarDecl*⌉* ⌈*Instruction*⌉* }

LocalVarDecl ⇒ *LocalType Id* **;**

LocalType ⇒ *Type* | **dir**

Instruction ⇒ *Assignment*

Assignment ⇒ *Id* = *Expr* **;**

Expr ⇒ *IntConstant* | *DoubleConstant* | *BoolConstant*
| (*Expr*) | *MonOp Expr* | *Expr BinOp Expr*
| *Id* (⌈*Expr* ⌈ **,** *Expr*⌉* ⌉)
| *Id* ⌈ [*Id* ⌈^ *Id*⌉*] ⌉

Fig. 7. CAOS state transition function syntax

much resembles the declaration of cell attributes in the cell section, the meaning is quite different. Cell attributes form the basis of the cellular automaton, are recomputed in each simulation step and can be read be neighbouring cells. In contrast, local variables in the behaviour section are not more than symbolic placeholders for intermediate values; their meaning is strictly limited to one instantiation of the state transition function.

Like the cell attributes, a local variable can be of any of the built-in types or of any of the previously defined enumeration types. What again separates local variables from cell attributes is the existance of one more built-in type: the direction type `dir`. The values of this type are the direction identifiers introduced in the grid construct. The only operations available on direction values are equality, inequality and the concatenation operator hat. With these restrictions direction values are first-class citizens of CAOS.

Left hand side variables in assignments can be either cell attributes or local variables. In the former case the new value of that attribute is defined; in the latter case the assignment has no effect, unless the local variable occurs in a subsequent expression that actually defines an attribute. Repeated assignment to the same cell attribute or local variable simply overwrites the previous value.

Right hand side expressions are made up of identifiers, the usual unary and binary operator applications and function applications. CAOS supports all built-in operators of C and the same associativities and priorities as in C apply. As CAOS in its current state does not support the definition of functions, function applications refer to external functions declared in the declaration section using the `extern` keyword.

Identifiers in expression position may either refer to cell attributes or to local identifiers. The latter case requires a preceding assignment to the local variable, otherwise its value and, hence, the value of the expression is undefined. If an identifier refers to the cell state, the value of the attribute is always the previous iteraions's value, even if an assignment to the same cell attribute precedes the occurence of the identifier.

A specialty of CAOS is the symbolic definition of neighbourhoods, more precisely the way the previous iteration's state of neighbouring cell's attributes are accessed. While a plain cell attribute identifier refers to the previous value of this attribute in the current cell, neighbouring cells can be accessed through symbolic selectors that make use of the direction identifiers defined together with the grid layout in the grid definition section. To support complex neighbourship relationships, multiple direction specifiers can be combined using the hat operator.

```
grid:X:100:left<.>right:cyclic, Y:100:up<.>down:cyclic;

cell {
  double state;
}

behaviour {
  double sum;
  sum = state + state[up^left] + state[up] + state [up^right]
              + state[left] + state[right]
              + state[down^left] + state[down] + state [down^right];
  state = sum / 9.0;
}
```

Fig. 8. CAOS example comuting average with eight neighbouring cells

Fig.8 illustrates the use of selectors using a simple CAOS program, where the state is a made up of a floating point number, the grid is two-dimensional with cyclic boundary conditions, and the state transition function computes the arithmetic mean between previous value and those of the eight neighbouring cell's values.

2.6 Control flow constructs

Simple state transition functions made up of sequences of assignments are too restricted to define most interesting cases. Even Conway's Game of Life, despite all its simplicity, cannot defined in this restricted setting. For general-purpose applicability CAOS supports a number of control flow manipulating constructs, some of which are directly borrowed from other languages (more precisely from C as far as syntax is concerned), some are tailor-made for CAOS. Fig. 9 defines the additional syntax for CAOS behaviour sections.

Instruction	⇒	*Assignment* | *Cond* | *Switch* | *ForEach*
Block	⇒	*Instruction*
	|	{ ⌈*Instruction*⌉* }
Cond	⇒	**if** (*Expr*) *Block* **else** *Block*
Switch	⇒	**switch** (*Id*) { ⌈*Case*⌉+ ⌈*Default*⌉ }
Case	⇒	**case** *CaseVal* ⌈**,** *CaseVal*⌉* ⌈*Guard*⌉ **:** *Block*
Guard	⇒	| *Expr*
Default	⇒	**default** **:** *Block*
ForEach	⇒	**foreach** (*LocalType Id* **in** *Set* ⌈*Guard*⌉ **)** *Block*
Set	⇒	[*Expr* ⌈**,** *Expr*⌉*]

Fig. 9. CAOS control flow construct syntax

The simplest control flow construct is a conditional or branching construct as supported in one way or another by any programming language. Fig. 10 illustrates its use and shows how Conway's Game of Life can be implemented with just this control flow construct.

```
cell {
  dead_or_alive state;
}

behaviour {
  int counter;

  counter = 0;
  if (state == alive)        counter=counter+1;
  if (state[up] == alive)    counter=counter+1;
  if (state[down] == alive)  counter=counter+1;
  if (state[left] == alive)  counter=counter+1;
  if (state[right] == alive) counter=counter+1;

  if (counter == 2 || counter == 3) {
    state = alive;
  }
  else {
    state = dead;
  }
}
```

Fig. 10. CAOS example implementing the Game of Life behaviour

The second control flow construct is a `switch`-construct. Despite the obvious syntactic similarities, the CAOS `switch` does differ from its C counterpart in essentially two aspects. Each `case`-statement may feature multiple values rather than just one as in C. Accordingly, CAOS does not feature multiple cases sharing the same block of instructions as would be achieved in C by leaving out the `break`-statement in between. Having said that, CAOS does not know the `break`-statement and it will always execute the block of instructions associated with the first case that fits the pattern as defined by the variable following the key word `switch`. The other difference between CAOS and C is that individual cases can be refined by a *guard* expression, which is syntactically is separated from the values by a vertical bar. If the switch variable has one of the values listed by some case, the guard expression is evaluated. This expression must yield a Boolean value. If the value is `true`, the associated block of instructions is executed; otherwise, the execution of the `switch`-construct continues with the next case.
CAOS does not feature any C-like loop constructs, but it does have a related construct, named `foreach`. The `foreach`-construct allows the programmer to define a block of instructions that is executed for each element of a given *set* of elements. Each expression of the set (enclosed by square brackets) is assigned exactly once to the identifier introduced in the construct and the associated block of instructions is executed for this value. As Fig. 9 shows, the set definition may be followed by an optional guard expression that has the same meaning as in the `switch`-construct introduced before. Fig. 11 illustrates the use of `foreach` and `switch` through an alternative implementation of the Game of Life.

2.7 Non-deterministic features

In some scenarios it is desirable to introduce probabilistic behaviour of cells. For this purpose three constructs `forone`, `with` and `choose` are provided as part of the CAOS language. Fig. 12 defines their exact syntax.
The `forone` constructs syntactically very much resembles the (deterministic) `forall`-construct. However, unlike `forall`, `foreach` selects exactly one element from the given set. The element is selected using pseudo-random methods. Once an element from the set is chosen, the (optional) guard expression is evaluated. If it evaluates to `true`,

```
cell {
  dead_or_alive state;
}

behaviour {
  int counter;

  counter = 0;

  foreach (dir d in [left, right, . , up, down] | state[d] == alive) {
    counter = counter + 1;
  }

  switch (counter) {
    case 0,1,4: state = dead;
    case 2,3:   state = alive;
  }
}
```

Fig. 11. CAOS alternative example implementing the Game of Life behaviour using advanced control flow constructs

Instruction ⇒ ...
| *ForeOne* | *With*

ForOne ⇒ **forone** (*Type Id* **in** *Set* ⌈*Guard*⌉) *Block*

With ⇒ **with** (*Expr*) *Block* ⌈ **else** *Block* ⌉

Expr ⇒ ...
| *Choose*

Choose ⇒ **choose** (*LocalType Id* **in** *Set*)

Fig. 12. CAOS syntax of non-deterministic language constructs

the associated block of instructions is executed; otherwise, program execution proceeds to the instruction following the `forone`-construct.
Fig. 13 illustrates the use of the `forone`-construct (and of the other non-deterministic langauge features introduced below). In the first behaviour section one of the four direct neighbours is non-deterministacally chosen and the state of that neighbour defines the new state of the current cell.
If the sole intention of using a `forone` is to non-deterministically choose one value out of a set of values, the `choose`-construct provides a more concise alternative. The `choose`-construct actually is an expression rather than an instruction. It can be used anywhere in expression position; its value is one of the values described by the set, which one is non-deterministc. The second behaviour section in Fig. 13 illustrates the use of `choose`.
Last not least, the `with`-construct introduces the notion of probabilistic execution of code blocks. After specifying a probability $0 \leq p \leq 1, p \in \mathbb{R}$ a block of code will be executed with this probability. With probability $1 - p$ the code block following the key word `else` is executed. The absence of an `else`-block is treated as an empty `else`-block. The third behaviour block in Fig. 13 illustrates the use of `with`. With 70% propability the state of the left neighbour is chosen as new state of the current cell, with 30% propability the state of the right neighbour.

```
behaviour {
  forone (dir d in [left, right, up, down]) {
    state = state[d];
  }
}

behaviour {
  state = choose (int val in [state[left], state[right], state[up], state[down]);
}

behaviour {
  with (0.7) state = state[left];
  else state = state[right];
}
```

Fig. 13. Examples for the use of CAOS non-deterministic features

2.8 Observers

It is paramount for any simulation software to make the result of simulation, and in most cases intermediate states at regular intervals as well, visible for interpretation. Observers serve exactly this purpose. They allow us to observe the values of certain attributes of cells and agents or cumulative data about them (e.g. averages, minima or sums) at certain regular intervals of the simulation or just after completing the entire simulation.
Each observer is connected with a certain file name (not a certain file). The parallel runtime system takes full advantage of parallel I/O both when using MPI and OPENMP as backend. This file system handling is particularly tricky if it is to be hand-coded. An auxiliary tool suite provides a comfortable user-interface to observer data produced through parallel file I/O.

Observer	⇒	*Observeall* | *Observe*
Observeall	⇒	**observeall** (*Filename* **,** *Expr*) { ⌈*ObsAllInstr*⌋+ }
Observe	⇒	**observe** (*Filename* **,** *Expr*) { ⌈*ObsInstr*⌋+ }
ObsAllInstr	⇒	*Type String* = *Expr* **;**
ObsInstr	⇒	*Type String* = *ReduceOp* (*Expr*) **;**
ReduceOp	⇒	**avg** | **min** | **max** | **sum** | **prod** | **all** | **any** | **cnt**

Fig. 14. CAOS observer syntax

There are two conceptual different classes of observers (see Fig. 14 for concrete syntax) that either observe values of cell attributes individually or that apply a reduction operation to all cells and produce scalar results. Both concepts allow the programmer to specify time steps in which the blocks are executed. For this, an Boolean expression depending on the current `timestep` may be specified. It is evaluated in each time step. If the value is true, the associated observation block is executed. Both observer classes require the declaration of a filename. All data gathered by the observer are written to the file specified by that filename. Each entry in an observation block is build up in the same way: First, the type of the result has to be specified, followed by a user-definable identifier of that entry. The identifier is followed by the expression that computes the desired result.

```
observe ("myObserverFile", timestep%10 == 0) {
  int "cellsAlive" = cnt( state == alive);
}

obserevall ("mySnapshot", timestep == 1) {
  int "cellState" = state;
  bool "activeState" = active;
}
```

Fig. 15. Examples for the use of observers

The keyword `observe` denotes the start of an observer that uses reduce operations on selected components. Each line in the observe section yields one value that is written to file. There are eight different reduce operators available:

- `min/max/avg` These operators determine the minimum/maximum/average of the given expression for all cells.
- `sum/prod` The value of the expression for each cell is summed/multiplied up.
- `cnt` If the Boolean expression holds, a counter is increased by one. Eventually, the value of the counter is written to the file.
- `any/all` These operators yield true if the expression is true for all cells/any cell and false otherwise.

Fig. 15 illustrates the use of observers by two simple examples. The first observer observes the number of cells that are in state `alive` after every 10 timesteps.
The keyword `observeall` starts the definition of an observer that saves values for every single cell to the file. The resulting file contains a tuple of results for every cell, representing the results of expressions given in the observer section. Thus, a simple snapshot of the grid is generated by specifying the cell components as results in the observer section in the same order as they appear in the `cell` section. Fig. 15 again provides a simple example.

2.9 Agents

Agents are similar to cells in that they consist of a set of attributes. Agents move from cell to cell; at any step during the simulation an agent is associated with exactly one cell. A cell in turn may be associated with a conceptually unlimited number of agents. Like the cells, agents have a behaviour (or state transition function). The behaviour of an agent is based on its existing state and the state of the cell it resides at as well as all other agents and cells in the neighbourhood as described above. In addition to updating its internal state, an agent (unlike a cell) may decide to move to a neighbouring cell. Conceptually, this is nothing but an update of the special attribute location. Agents also have a life time, i.e. rather than moving to another cell, agents may decide to die and agents may create new agents. Agents are not implemented in the current version of CAOS but are planned for the next release.

3. The CAOS tool-chain

We have implemented a fully fledged CAOS compiler[1] that generates sequential C code. On demand, the grid is automatically partitioned for multiple MPI processes. The process topology including the choice and number of partitioned grid axes are fully user-defined. A default process topology provided at compiler time may be overwritten at program startup. Additionally, each MPI process may be split either statically or dynamically into a

[1] The current version does not yet support agents.

user-defined number of OPENMP threads, provided that the available MPI implementation is thread-safe. Proper and efficient communication between MPI processes including the organisation of halo or ghost cells at partition boundaries is taken care of by the compiler without any user interaction.
The compilation of a CAOS simulation into an executable binary is based on the CAOS compiler *caosC*. Invocation of the compiler is done indirectly via a generic Makefile that implements all required stages from translating CAOS source code to C, choosing an appropriate C compiler and calling the desired compiler with all required options to generate binary code.
The compilation process is parametrised over several options that determine what kind of parallelisation is applied and how many time steps are executed by default. If `make` is called without any options, an extensive help screen is displayed.
The resulting program carries out the simulation steps as shown in Fig. 16. As with

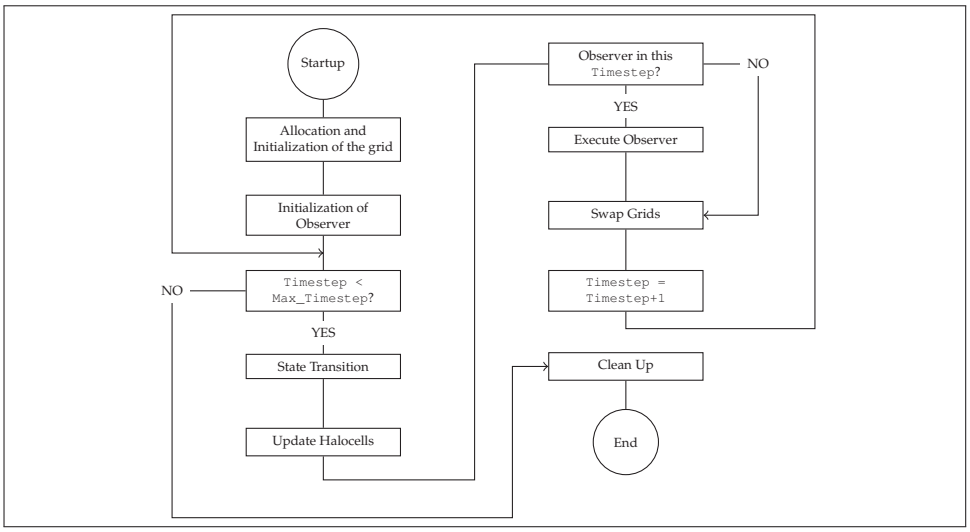


Fig. 16. Program flow chart of a CAOS simulation run

the compilation process, the executable file supports several options too. The parameters influence the runtime behaviour of the program and are as follows:

- `-help` displays a list of all supported parameters
- `-timesteps=`*n* sets the number of simulation time steps to *n*. If this parameter is not given, the compiled-in default number of time steps is executed.
- `-show-progress` shows a visual progress indicator during execution.
- `-infile=`*filename* if this option is present, the initial state of the cell grid is read in from a file *filename*. The `init` block of the CAOS program has no effect in this case.
- `-dimsizes=`*N* this option overrides the compiled-in grid size of the simulation. When using this option, the sizes of all dimensions of the grid have to be specified. Sizes are separated by a lower-case `x`, for example 128 `x` 128.
- `-psizes=`*P* defines the distribution of the grid for each dimension. With this parameter each dimension is divided into as many parts as defined by *P*.Sizes are separated by a

lower-case `x`, for example 2 `x` 2. If a dimension is given a size of 1 no distribution along that dimension is applied. See Fig. 17 for examples.

- `-mpi-psizes=`*P* for simulations that have been compiled into mixed-mode binaries, i.e. simultaneous usage of OpenMP and MPI, this parameter defines the distribution of the grid across MPI processes for each dimension. Each dimension is divided into as many parts as defined by *P*.Sizes are separated by a lower-case `x`, for example 2 `x` 2. If a dimension is given a size of 1 no distribution along that dimension is applied.
- `-omp-psizes=`*P* for simulations that have been compiled into mixed-mode binaries, i.e. simultaneous usage of OpenMP and MPI, this parameter defines the distribution of the grid across OpenMP threads *per MPI process* for each dimension. Each dimension is divided into as many parts as defined by *P*.Sizes are separated by a lower-case `x`, for example 2 `x` 2. If a dimension is given a size of 1 no distribution along that dimension is applied.
- `-print-defaults` print compiled-in defaults of all settings
- `-setparam:`*Q*=*v* initializes parameter *Q* with value *v*. The parameters that can be set here are those that have been specified in the CAOS source code using the `param` keywork. A list of all compiled-in parameters is displayed within the help text of the binary (`-help` option).

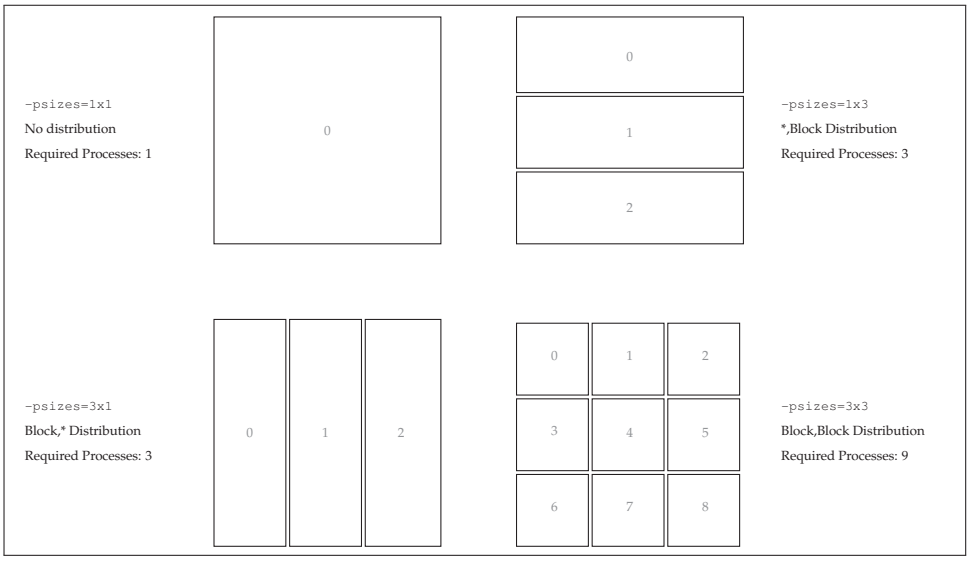


Fig. 17. Examples of possible distributions of a two-dimensional grid using the `-psizes` parameter

For a more detailed treatment of the semantics of these parameters and for an in-depth discussion of all implementation details, please see Grelck & Penczek (2007).

4. Selected implementation aspects

The main component of the CAOS tool-chain is the compiler that infers all static information of a CAOS program and compiles many default settings into the binary file. Still, certain

dynamic aspects of a simulation are determined at runtime when a simulation is run with values other than the compile-time defaults. For parallel execution, this includes the distribution of the cell grid and appropriate communication patterns. An overview of the general execution of a parallel CAOS program is shown in Fig. 18. We will describe a few

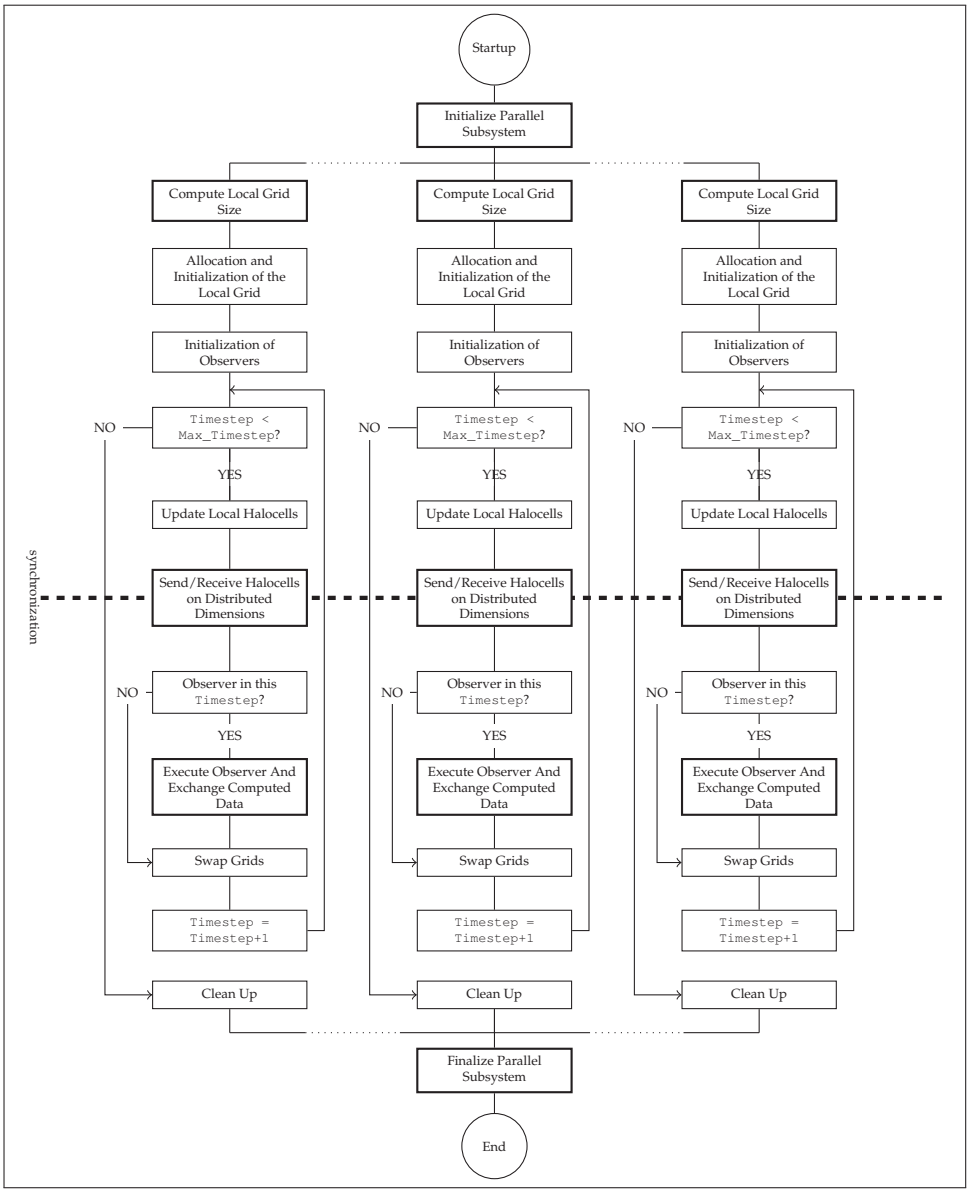


Fig. 18. Program flow plan of a parallel CAOS program

selected aspects of this process in more detail in the following sections.

4.1 Halo-cells: Inference, distribution and communication

In an implementation of a cellular automata simulation the cell grid is inevitably bounded in size, i.e. there will be certain cells that live on the boundary of the grid. If a dimension is defined as being cyclic, the neighbouring cell at the boundary lies at the other end of the grid. If the boundary is a static one the cells does not have a real neighbour. To avoid out-of-bounds errors without changing the access pattern of boundary cells the grid has to be padded as shown in Fig. 19 so that accesses to all neighbours can succeed.

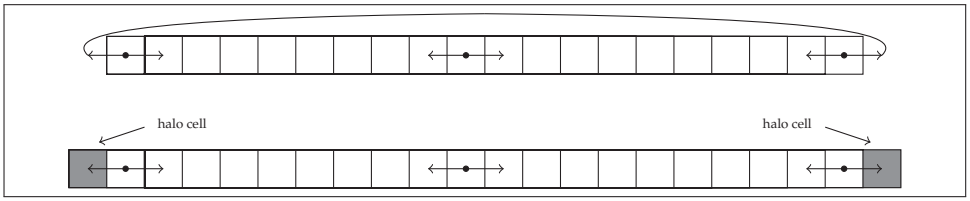


Fig. 19. Access to adjacent cells at cyclic (top) and static boundaries (bottom). At static boundaries halo-cells are automatically introduced.

The padding of the original cell grid introduces a frame of halo cells. This frame extends the original cell grid on each side by as many cells as are required for all accesses within a `behavior` to stay within the framed grid. An automatic inference mechanism analyses access patterns and determines the minimum size of the halo frame. Generally, this will lead to different extensions of dimensions as shown in Fig. 20. Halo cells are built from the same

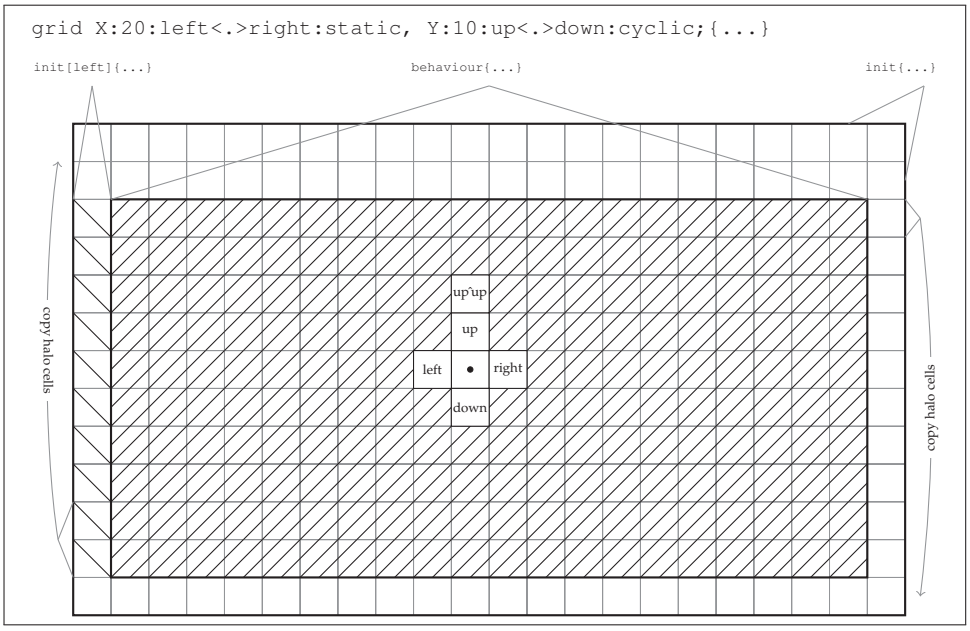


Fig. 20. Automatic embedding of the cell grid into a frame of halo cells. The extend of the frame is automatically inferred.

constituents as standard cells as defined in the `cell` block of a CAOS program. Consequently,

the initial values of halo cells are set by the `init` block. However, if more control over these initial values is required, `init` blocks for all or some of the halo frames may be given in additional `init` blocks. Fig. 20 shows this for a two-dimensional grid where the left side of the halo frame is initialised with special values.
Halo cells do not perform state transitions, they remain static during the entire simulation. If a grid dimension is defined as being cyclic, however, the halo cells are automatically updated after each time step so that cells on boundaries access the correct values of cells that are located on the other end of the grid. Because of the halo cell frame, these accesses are just applications of the standard access pattern and do not require complex index transformations.
In a parallel setting, where the global grid is divided up into several smaller grids, the halo frames extend each local grid. Along dimensions that are not distributed the halo cells serve the same purpose as before. Along dimensions that are distributed, however, the halo cells are used to represent cells of neighbouring local grids. This ensures that distribution is completely transparent for the rest of the implementation, especially the implementation of the state transition function. During each time step of the simulation the values of halo cells are exchanged with the appropriate adjacent neighbour of a distributed dimension as shown on Fig. 21.

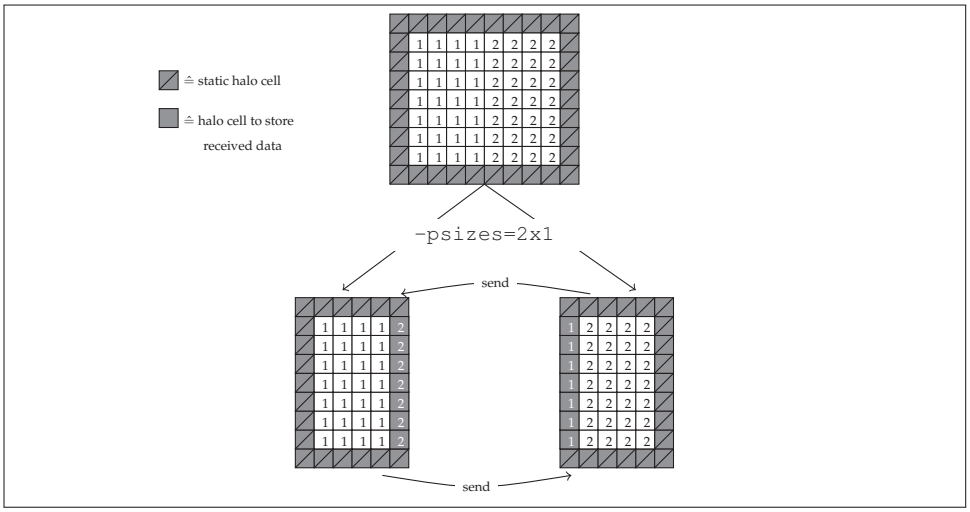


Fig. 21. Halo cells serve a dual purpose on a distributed grid. Along distributed axises the halo frame keeps copies of boundary cells of neighbouring local grids.

4.2 Parallel I/O

Writing data during the execution of observers poses a challenge when a simulation is executed in parallel. As the global grid does not exist as a whole, but is distributed over several processes, several concurrent write operations have to be carried out to the same file. Of course, this could be avoided by collecting all local grids in one process and then executing the observer code on a completely reassembled grid. Obviously, this procedure requires a considerable amount of data to be send and it may also require more memory than a single process has available to store the grid. The CAOS implementation takes a different approach where each process writes its local grid directly to the appropriate location within a shared file.

Each process uses information about the extent of the global grid and the distribution of the grid across several processes. From this information a process computes the location of the part of the global grid that it locally holds. This determines which regions of a file the process will fill with the contents of its local grid during the execution of an observer.
The MPI standard defines a set of I/O functions that offers high-performance I/O operations in a distributed setting on a high level of abstractions. For CAOS we make use of the *file view* concept Gropp et al. (1999), which uses information about the global grid and the information about the distribution to organise concurrent access to files into independent tasks. The file view determines which (not necessarily consecutive) parts of a file are visible to a process. The remaining parts of the file that belong to other processes are masked out. Using this, each process may apply its `observe` blocks in the same way as for sequential execution as the MPI I/O functions automatically direct file access to the appropriate location of the observer file. See Fig. 22 for an illustration.

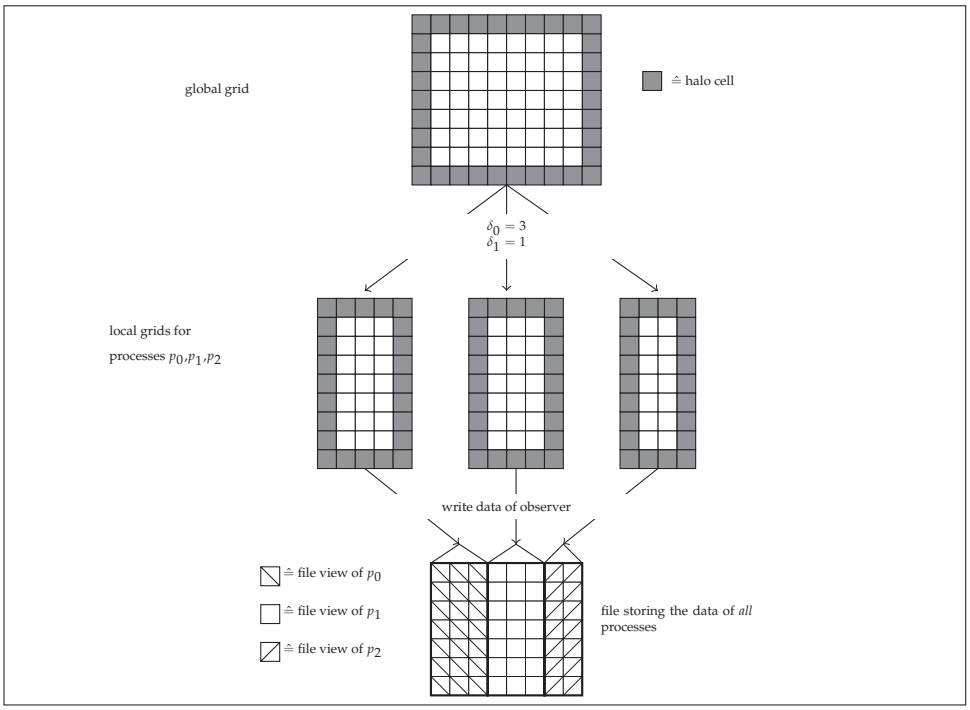


Fig. 22. Using MPI's dedicated high-performance I/O functionality, local *file views* are created for each local grid.

5. Evaluation and performance

We use an implementation of a 2-dimensional Jacobi iteration as the basis for performance evaluation experiments. Fig. 23 shows the complete CAOS code, which also serves as a reference example for CAOS programs.

We measured the runtime of this program on several machines to cover multiple common hardware configurations: a dual-core laptop, a cluster of distributed-memory blade servers and a 48-core shared-memory computation server.

```
param int atTstep = 1;
param int size = 100;

grid 0..size : left <.> right : static,
     0..size :   up <.> down  : static;

cells { double state; }

init { state = 0.0; }

init[down] { state = 500.0; }

behaviour { double a = 0.0;
            foreach (dir d in [up,down,left,right]) {
              a = a + state[d];
            }
            state = a / 4.0;
          }

observeall ("jacobi.outfile.all", timestep==atTstep) {
  double "state" = state;
}

observe ("jacobi.outfile.reduce", timestep==atTstep) {
  double "avgState" = avg(state);
}
```

Fig. 23. Jacobi iteration specified in CAOS

5.1 Performance on consumer-grade hardware

As a representative measurement for consumer-grade hardware, we have measured the Jacobi iteration on a 4096×4096 grid for 1000 timesteps. The laptop runs Mac OS X 10.6.4 on a 2.4GHz Intel Core 2 Duo and contains 4GB of main memory. The simulation has been compiled using the OpenMP back-end of the CAOS compiler. The generated code exploited both cores when run with two threads and achieved a speed-up of almost 1.8 as Fig. 24 shows.

5.2 Performance on distributed memory

The cluster consists of nodes with 2 E5520 Intel Xeon processors with hyperthreading disabled. Each node contains 24GB of ram, network connections between the nodes are established via DDR Infiniband. All nodes have access to a shared file system.
The runtimes shown in Fig. 25 are based on the execution of the Jacobi iteration on a 16384×16384 grid for 1000 time steps. Although the effect diminishes with increasing numbers of MPI processes, the simulation runtimes decrease with the number of available computing resources.

5.3 Performance on shared memory

The machine we have used for these runs is a 48-core shared-memory system consisting of 4 twelve-core AMD Opteron 6174 processors. The total amount of memory in the system is 256GB to which the cores have non-uniform access.
We ran two series of experiments, both using the Jacobi implementation on a grid of 16384×16384 for 1000 time steps. The first series of experiments used the OpenMP back-end of the CAOS compiler, the second series was distributed using MPI. As Fig. 26 shows, both versions

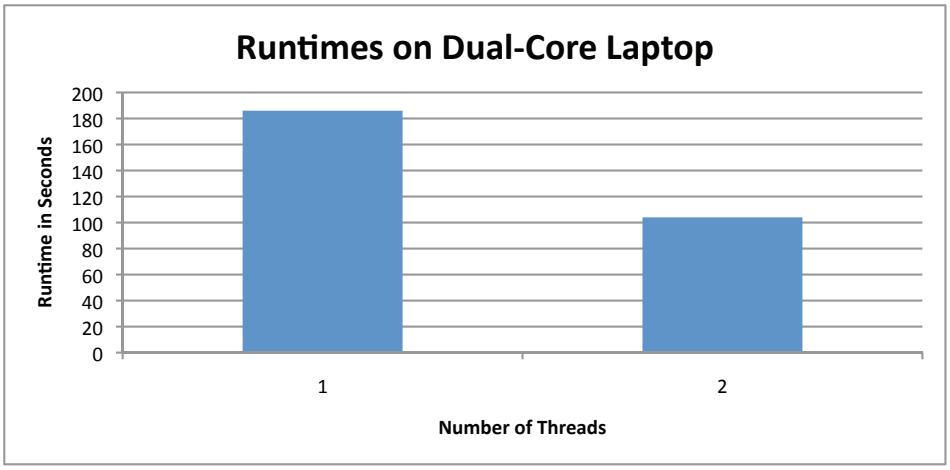


Fig. 24. Runtimes on a standard off-the-shelf laptop.

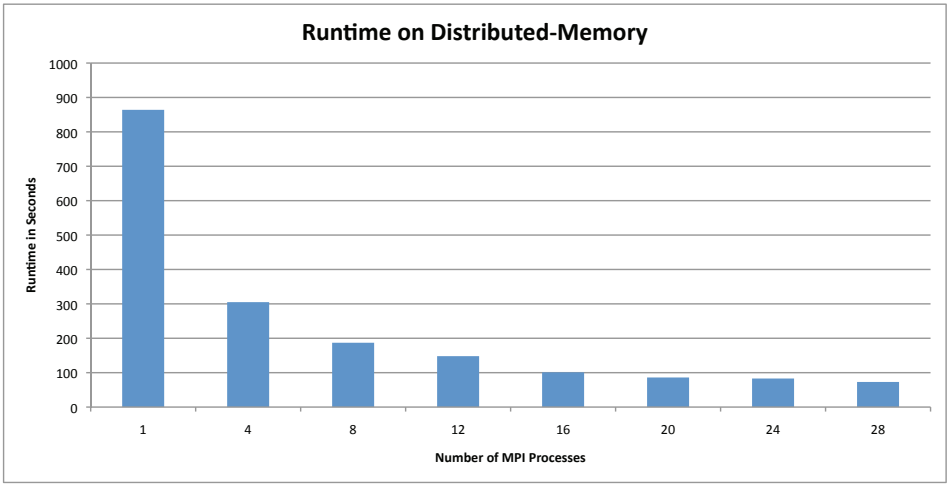


Fig. 25. Runtimes on a distributed-memory cluster using MPI

scale for a small number of cores. The OpenMP version is more efficient than the MPI variant for a small numbers of cores. However, on this machine the OpenMP implementation did not scale beyond 6 cores for reasons that would require further investigation. The MPI version did not suffer from this problem and scaled with the number of cores which reduced the runtime of the simulation considerably.

6. Related work

Mathematica and MatLab are well-known general-purpose systems that are also suitable for implementing cellular automata on a level of abstraction that exceeds that of standard

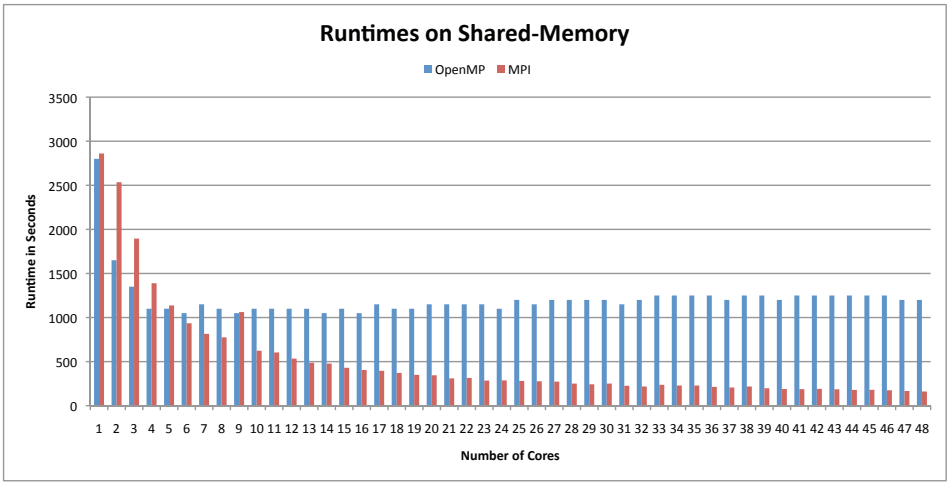


Fig. 26. Runtimes on a 48-core shared-memory machine using MPI and OpenMPI

programming languages. However, when it comes to complex simulations it is often more convenient to use a domain-specific, high-level language to implement the simulation. A programmer may choose from a range of languages like CANL Calidonna & Furnari (2004), CDL Hochberger et al. (1995), TREND Chou et al. (2002) and JCASim Freiwald & Weimar (2002). These languages offer an instruction set that is specifically tailored towards cellular automata. The simulations, however, are restricted to two-dimensional automata. Languages like CARPET/CAMEL Spezzano & Talia (1997a;b) and CELLANG Eckart (1992) overcome this restriction and offer support for automata with one, two and three dimensions. CAOS takes the idea even further and supports an arbitrary amount of dimensions.

The parallel execution of simulations is for example supported by CARPET Spezzano & Talia (1997a) with the CAMEL system Naumov (2004). CAOS supports the multi-threaded execution on shared memory machines and a distributed execution on clusters. If cluster nodes are multi-processor, shared-memory machines, CAOS also supports the combination of both models. Processes are distributed over the cluster nodes where the execution of each process is multi-threaded.

The regular structure of grids of cellular automata makes them well-suited for a direct mapping onto reconfigurable hardware. In Halbach et al. (2004) the simulation of cellular automata on FPGAs is investigated; in Ackermann et al. (2001) the design of a pseudo-random number generator in hardware on the basis of a CA is described. Compilers that translate cellular automata programs to these platforms are available, as for example the CDL compiler Hochberger et al. (1995).

7. Conclusion

CAOS is a new domain-specific programming language for the high-level specification of numerical simulations based on the well-known concept of cellular automata. CAOS extends this concept in a number of directions. For instance, grids are not limited to vectors or matrices, but can actually have any number of dimensions/axes. Communication is not restricted to nearest neighbours, but may cover any (static) neighbourhood. Cells do not carry binary information, but aggregate any number of numerical properties or attributes

as required by the programmer. Last not least, the state transition function is defined by means of a simple but nevertheless fully-fledged programming language whose design is geared towards the given purpose and, in particular, features a number of non-deterministic constructs.

Our CAOS compiler exploits the restricted pattern of communication characteristic for cellular automata for generating executable code whose runtime performance is highly competitive on modern computer architectures. Fully automatic parallelisation for shared memory architectures based on OpenMP as well as for distributed memory architectures based on MPI provides easy access to high-performance computing infrastructures from state-of-the-art symmetric multiprocessors to clusters of workstations and supercomputers. All this can be harnessed with only modest (sequential) programming skills and practically no familiarity with modern computer architecture or parallel computing issues.

We currently pursue two directions of future work. Firstly, we plan to continue on the successful route to support compiler-directed parallelisation through a restricted model of computation and extend the CAOS compiler to support emerging architectures such as general purpose graphics processors. Secondly, we are working on completing the CAOS language by support for agents that substantially increase the expressiveness of CAOS for advanced simulation.

Further information on the CAOS project, including a technical report that covers compilation in-depth Grelck & Penczek (2007) and a source distribution with demos for download, is available at

`http://www.caos-home.org/` .

8. References

Ackermann, J., Tangen, U., Bödekker, B., Breyer, J., Stoll, E. & McCaskill, J. (2001). Parallel random number generator for inexpensive configurable hardware cells, *Computer Physics Communications* 140: 293–302.

Amarasinghe, S. (2008). (How) can Programmers Conquer the Multicore Menace?, *17th International Conference on Parallel Architectures and Compilation Techniques (PACT'08)*, ACM, New York, NY, USA, pp. 133–133.

Boccara, N., Goles, E., Martinez, S. & Picco, P. (eds) (1993). *Cryptography with Dynamical Systems*, Kluwer Academic Publishers, pp. 237–274.

Calidonna, C. & Furnari, M. (2004). The cellular automata network compiler system: Modules and features, *International Conference on Parallel Computing in Electrical Engineering*, pp. 271–276.

Canyurt, O. & Hajela, P. (2005). A cellular framework for structural analysis and optimization, *Computer Methods in Applied Mechanics and Engineering* 194: 3516–3534.

Chapman, B. (2007). The Multicore Programming Challenge, *Advanced Parallel Processing Technologies*, p. 3.

Chou, H., Huang, W. & Reggia, J. A. (2002). The Trend cellular automata programming environment, *SIMULATION* 78: 59–75.

Conway, J. (1970). The game of life, *Scientific American* .

Dagum, L. & Menon, R. (1998). OpenMP: An Industry-Standard API for Shared-Memory Programming, *IEEE Transactions on Computational Science and Engineering* 5(1).

D'Ambrosio, D., Iovine, G., Spataro, W. & Miyamoto, H. (2007). A macroscopic collisional model for debris-flows simulation, *Environmental Modelling & Software* 22: 1417–1436.

Eckart, D. (1992). A cellular automata simulation system: Version 2.0, *ACM SIGPLAN Notices* 27.

Ermentrout, G. B. & Edelstein-Keshet, L. (1993). Cellular automata approaches to biological modeling, *Journal of Theoretical Biology* 160: 97–133.

Freiwald, U. & Weimar, J. (2002). The Java based cellular automata simulation system JCASim, *Future Generation Computing Systems* 18: 995–1004.

Gabb, H., Mattson, T. & Breshears, C. (2009). Thinking in Parallel - Three engineers' Viewpoints, *Intel Software Insight Magazine* 16: 24–26.

Georgoudas, I. G., Sirakoulis, G. C., Scordilis, E. M. & Andreadis, I. (2007). A cellular automaton simulation tool for modelling seismicity in the region of Xanthi, *Environmental Modelling & Software* 22(6): 1455–1464.

Grelck, C. & Penczek, F. (2007). CAOS: A Domain-Specific Language for the Parallel Simulation of Extended Cellular Automata and its Implementation, *Technical report*, University of Lübeck, Institute of Software Technology and Programming Languages.

Gropp, W., Lusk, E. & Skjellum, A. (1994). *Using MPI: Portable Parallel Programming with the Message Passing Interface*, MIT Press, Cambridge, Massachusetts, USA.

Gropp, W., Lusk, E. & Thakur, R. (1999). *Using MPI-2*, The MIT Press, chapter Parallel I/O, pp. 60–64.

Guisado, J., de Vega, F. F., Jiménez-Morales, F. & Iskra, K. (2006). Parallel implementation of a cellular automaton model for the simulation of laser dynamics, *LNCS* 3993: 281–288.

Halbach, M., Hoffmann, R. & Röder, P. (2004). FPGA implementation of cellular automata compared to software implementation, *Organic and Pervasive Computing*, Lecture Notes in Informatics, pp. 309–317.

Hochberger, C., Hoffmann, R. & Waldschmidt, S. (1995). Compilation of CDL for different target architectures, *Parallel Computing Technologies*, Vol. 964 of *LNCS*, Springer, pp. 169–179.

Nagel, K. & Schreckenberg, M. (1992). A cellular automaton model for freeway traffic, *J. Phys. I France 2* .

Naumov, L. (2004). CAME&L – cellular automata modeling environment and library, *LNCS* 3305: 735–744.

Popovici, A. & Popovici, D. (2002). Cellular automata in image processing, *Technical report*, University of the West Timisoara, Romania.

Spezzano, G. & Talia, D. (1997a). A high-level cellular programming model for massively parallel processing, *Proc. 2nd Int. Workshop on High-Level Programming Models and Supportive Environments (HIPS'97)*, IEEE Press, pp. 55–63.

Spezzano, G. & Talia, D. (1997b). Programming high performance models of soil contamination by a cellular automata language, *High-Performance Computing and Networking*, Vol. 1225 of *LNCS*, Springer, pp. 531–540.

Stevens, D., Dragicevic, S. & Rothley, K. (2007). iCity: A GIS-CA modelling tool for urban planning and decision making, *Environmental Modelling & Software* 22(6): 761–773.