Evolutionary computing & CA models

A genetic algorithm tool to optimize the Bayesian calibration of an urban land use change model

Cláudia M. ALMEIDA[1]; Britaldo S. SOARES-FILHO[2]; Hermann O. RODRIGUES[2]

[1]National Institute for Space Research – INPE, Division for Remote Sensing - DSR
Av. dos Astronautas, 1758, SERE – I, PO Box 515, 12227-010 São José dos Campos, Brazil
+55-12-3208-6428, almeida@dsr.inpe.br (correspondent author)
[2]Federal University of Minas Gerais – UFMG, Centre for Remote Sensing – CSR/IGC
Av. Antônio Carlos, 6627, 31270-900, Belo Horizonte, Brazil
+55-31-3409-5449, {hermann, britaldo}@csr.ufmg.br

Keywords: genetic algorithm, random allocation of change, Dinamica EGO, weights of evidence method, fuzzy similarity index

Introduction

The employment of evolutionary (or Darwinian) premises for automated problem solving is not new and dates back from the 1950s. Nearly a decade afterwards, three different interpretations of this approach started to be developed in parallel by three distinct researchers. Lawrence J. Fogel [1] in the US was the first one to introduce the concept of evolutionary programming. John Henry Holland [2], on his turn, called his method a genetic algorithm. In Germany, the domain of evolution strategies arose with Ingo Rechenberg and Hans-Paul Schwefel [3]. It was only in the beginning of the 1990s that these three areas were merged under one major field called evolutionary computing. Also at this time an alike fourth stream had emerged – genetic programming. In this way, evolutionary computing turned out to embrace the sub-areas of evolutionary programming, evolution strategies, genetic algorithms, and genetic programming.

The field of evolutionary computing has presented linkages with Artificial Life, especially since the 1990s, with the swarm-based computation and nature-inspired algorithms. Genetic algorithms in particular gained popularity with the work of John Holland [2]. According to [4], the increasing academic interest in this field led to meaningful advances in the computers processing capacity for practical applications, including the automatic evolution of computer programs. Evolutionary algorithms, as stated in [5], "are now used to solve multi-dimensional problems more efficiently than software produced by human designers, and also to optimise the design of systems".

As [6] reported, the use of genetic algorithms in cellular automata (CA) models started at the end of the 1990s with the work of [7]. Other works in the same line were produced, as in [8] and [9], which used GA for parameter estimation of complex urban dynamic models, as well as in [10], [11], [12], and [13], in which transition rules of CA models have been optimized by genetic algorithms (GA).

More recently, there has been a profusion of articles dealing with GA for calibration and optimization of urban CA models [14], [15], [16], [17].
In all above-mentioned cases, a binary approach (urban x non-urban) has been adopted. In a diverse way from the previously reported works, the purpose of this paper is to deal with the simulation of multiple urban land uses (e.g. residential, commercial, industrial, etc.) by means of a GA tool employed to optimize a Bayesian calibration of a CA urban land use change model.

GA fundamentals

As stated by [18], evolutionary algorithms form a subset of evolutionary computation in that they generally only involve techniques implementing mechanisms inspired by biological evolution such as reproduction, mutation, recombination, natural selection and survival of the fittest. In this process, there are two main forces that form the basis of evolutionary systems: Recombination and mutation create the necessary diversity and thereby facilitate novelty, while selection acts as a force increasing quality [18].
Genetic algorithms, in brief, are methods that simulate the processes of natural and genetic evolution through computational routines, aiming to solve optimization problems in situations where the search space is huge and conventional methods have demonstrated to be inefficient. GA are basically structured in an analogous way to the biological chromosomes, as initially exposed. The first step consists in the generation of a population of individuals, which are characterized by their chromosomes, corresponding to numerical values representing a possible solution to a given problem. During the evolutionary process, this population is evaluated, and each chromosome is awarded a grade that reflects its adaptation capacity to a certain environment. The fittest chromosomes are selected, and the least fit ones are discarded, in accordance with Darwinian laws. The selected individuals are subject to cross-over (recombination) and mutation, generating offspring to the next generation, which corresponds to a complete iteration of the genetic algorithm. This process is repeated until a satisfactory solution is found [19].
Cross-over basically consists in combining the genetic material of two individuals, generating two new descendents, which inherit the parents´ characteristics. In order to avoid the anticipated convergence of the genetic algorithm, it is necessary to conduct a mutation operation, introducing new regions in the solutions search space. Many aspects of such an evolutionary process are stochastic. Changed pieces of information due to recombination and mutation are randomly chosen. On the other hand, selection operators can be either deterministic, or stochastic. In the latter case, individuals with a higher fitness have a higher chance to be selected than individuals with a lower fitness, but typically even the weak individuals have a chance to become a parent or to survive [18].
In order to assess the quality of a candidate solution, an objective function is used. It provides to the genetic algorithm a measure of fitness of each individual belonging to the population [19]. The choice of an appropriate objective function is crucial for the success of the GA performance. A detailed explanation on the objective function (or fitness function) employed in this work is presented in the next section.

Application

Study Area and the GIS Database

The GA-optimized CA simulation model was applied to a medium-sized city, Bauru, located in the Midwest of São Paulo State, southeast of Brazil. The city comprised a total of 236,740 inhabitants in the initial time of simulation (1988), which increased to 309,531 inhabitants in 2000. In this period, the annual population growth rate was around 1.34%, and it was marked by the expansion of the existing residential areas together with the mushrooming of peripheral residential settlements, which have been mostly incorporated to the main urban tissue (Figure 1).

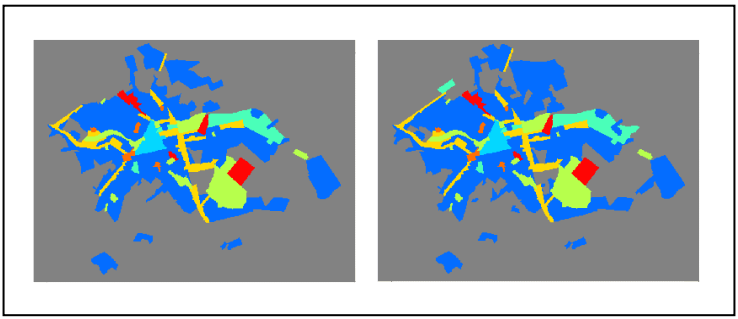

Figure 1: Land use map in Bauru in 1988 (left) and 2000 (right)

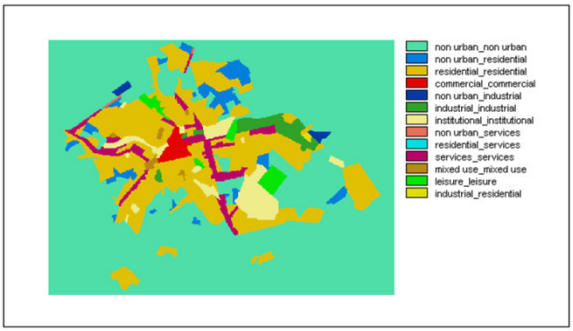

Figure 2: Cross-tabulation map between Bauru land use maps of 1988 and 2000, indicating permanence and changes in land use

Besides experiencing a considerable development concerning the residential use, Bauru also witnessed intra-urban land use changes like the increase in industrial and services areas (Figure 2).

The assessment of the total amount of land use change from 1988 to 2000, commonly known as global transition rates, was directly derived from a cross-tabulation operation between the initial and final land use maps, which provided the

figures presented in Table 1, associated with the five types of observed land use change.

Land Use	Non-Urban	Resid.	Comm.	Ind.	Inst.	Services	Mixed	Leis./ Recr.
Non-urban	0.9615	0.0333	0	0.0043	0	0.0009	0	0
Resid.	0	0.9997	0	0	0	0.0003	0	0
Comm.	0	0	1	0	0	0	0	0
Industr.	0	0.0438	0	1	0	0	0	0
Instit.	0	0	0	0	1	0	0	0
Services	0	0	0	0	0	1	0	0
Mixed	0	0	0	0	0	0	1	0
Leis./Recr.	0	0	0	0	0	0	0	1

Table 1: Global transition rates for Bauru: 1988–2000

After the identification of land use transitions and their respective rates, the next step concerned the determination of the different sets of infrastructure variables governing each of the five types of change, based on heuristic procedures. These procedures basically regard the visualization of distinct maps of variables (distances in grey scale) superposed on maps of land use transition, so as to identify those more meaningful to explain the different types of land-use change. The variables selected for modeling are listed in Table 2, and the sets of variables assigned to explain each of the five transitions are indicated in Table 3.

Notation	Physical or Socioeconomic Land Use Change Variable
dist_ind	Distances to industrial zones
dist_res	Distances to residential zones
dist_com	Distances to the central commercial zone
main_res	Distances to residential areas belonging to the main urban agglomeration
dist_serv	Distances to services corridors
serv_axes	Distances to the services and commercial axes
exist_rds	Distances to main existent roads

Table 2: Independent variables defining land use change in Bauru: 1988–2000

Notation	Nu_Res	Nu_Ind	Nu_Serv	Res_Serv	Ind_Res
dist_ind		♦			
dist_res	♦				
dist_com	♦	♦	♦		♦
main_res		♦	♦		
dist_serv				♦	
serv_axes		♦	♦	♦	♦
exist_rds	♦				

Table 3: Selection of variables determining land use change in Bauru: 1988–2000

All data used in this application had a resolution of 100 x 100 m and composed grids containing 487 lines and 649 columns, there being a total of 316,063 cells defining the region for simulation.

The GA-optimized Bayesian model of land use change

The GA-optimized Bayesian model of land use change was implemented in Dinamica EGO, a modeling environment that embodies neighborhood-based transition algorithms and spatial feedback approaches in a stochastic multi-step simulation framework. The parameterization method available at EGO is based on the theorem of conditional probabilities. For estimating the land use 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 use 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 use transition probability can be obtained through algebraic manipulations of the *logit* formula, as follows [20]:

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

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 use transition, e.g. from a class c to a class k, within a total of η transitions; 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; $O\ (T_i^{\alpha})$ 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., $P\ (T_i^{\alpha})/\ P\ (\overline{T_i^{\alpha}})$; 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 $P\ (V_i^{m_\alpha} / T_i^{\alpha})$ 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 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 land use transition 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 variables employed in the analysis, 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 categorization 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 land use change.

The GA tool in Dinamica EGO retrieves the W^{+} values and assembles them into tables. Each model parameter (W^{+}) represents an allele and will be a record in a table that corresponds to a gene. This group of tables is an input to the GA tool. GA tool spawns a population based on the genotype passed within a group of tables. Inside GA tool, a routine (or functor) called *Get current individual* is placed to get the genes from the individuals of a generation. Other functors are sequenced to get the parameters and pass them on to the model. An evaluation (fitness) function is coupled with the model output and its result is passed to a functor called *Set fitness*, which returns the fitness value to the GA tool for the selection process [21]. The internal sequence of functors will iterate a number of times as specified by the user.

When the GA tool terminates, it outputs the fitness of the overall best individual as well as the group of tables that comprises its genes. Additional parameters of the GA tool are: number of generations; population size; convergence stopping criteria, which forces the GA tool to terminate if evolution becomes asymptotical, as defined by the convergence limit, which must be achieved within the span of generations set by the number of generations; default lower and upper bounds, which set default values within which all allele values may vary; customized lower and upper bounds, defined by the user; amongst others [21].

The GA tool engine is based on the EO computation library and selects parental individuals for the next generation using one-to-one deterministic tournament. Individuals that take part in it are randomly drawn from the current population without depleting it. Cross-over creates 70% of the new generation individuals. Any allele in a gene of an individual chosen for mutation can be altered upon 1% probability. A new generation is completed by passing it the remaining 29% of parental individuals that were submitted to neither crossing-over nor mutation. Figure 3 illustrates the graphical user interface of the Dinamica EGO GA tool.

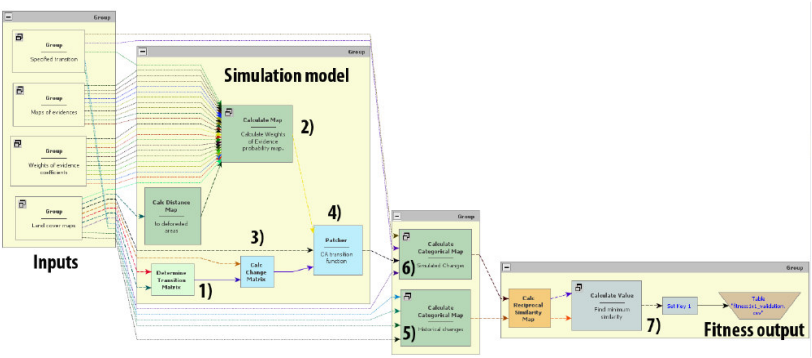

Figure 3: Graphical user interface of the GA model embedded in the Dinamica EGO

Objective Function and Validation

For assessing the fitness of the GA tool outputs as well as the accuracy of the CA simulation model performance, fuzzy similarity measures applied within a neighborhood context were used. The fuzzy similarity method employed in this work is a variation of the fuzzy similarity metrics developed by [21], and has been implemented in the Dinamica EGO platform.

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 neighborhood. Not considering fuzziness of category, the fuzzy neighborhood 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 uses), assuming 1 for a category = *i*, and 0 for categories other than *i*. Thus, the fuzzy neighborhood vector (*Vnbhood*) for each cell is given as:

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

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

where μ *nbhood i* represents the membership for category *i* within a neighborhood of *N* cells (usually $N=n^2$); μ *crisp* $_{i,j}$ is the membership of category *i* for neighboring 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 neighboring cell *j*, where *m* accounts for a distance decay function, for instance, an exponential decay ($m = 2^{-d/2}$). 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 [21]. 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 neighborhood 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 neighborhood 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 neighborhood vectors for maps *A* and *B*, and $\mu_{A,i}$ and $\mu_{B,i}$ are their neighborhood memberships for categories $i \subset C$ in maps *A* and *B*, as in equation 4. According to [22], since the similarity measure $S\ (V_A, V_B)$ tends to overestimate the spatial fit, the two-way similarity is instead applied:

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

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 use 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 Dinamica EGO team introduced a modification into the overall two-way similarity method of DINAMICA, 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 analyzed separately using pair-wise comparisons involving maps of differences: (i) between the initial land use map and a simulated one, and (ii) between the same initial land use 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 nulls 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.

Simulations and Discussion

The GA-optimized simulation and the land use change probabilities maps are respectively presented in Figures 4 and 5, demonstrating a good performance of the model.

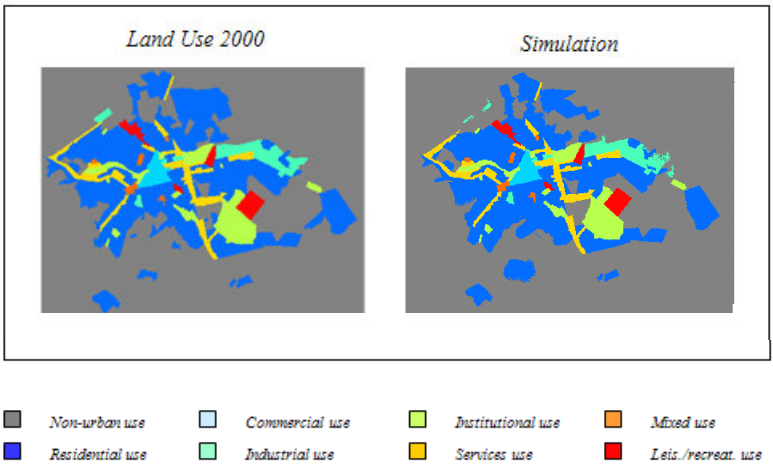

Figure 4: The GA-optimized simulation compared to the actual land use in 2000

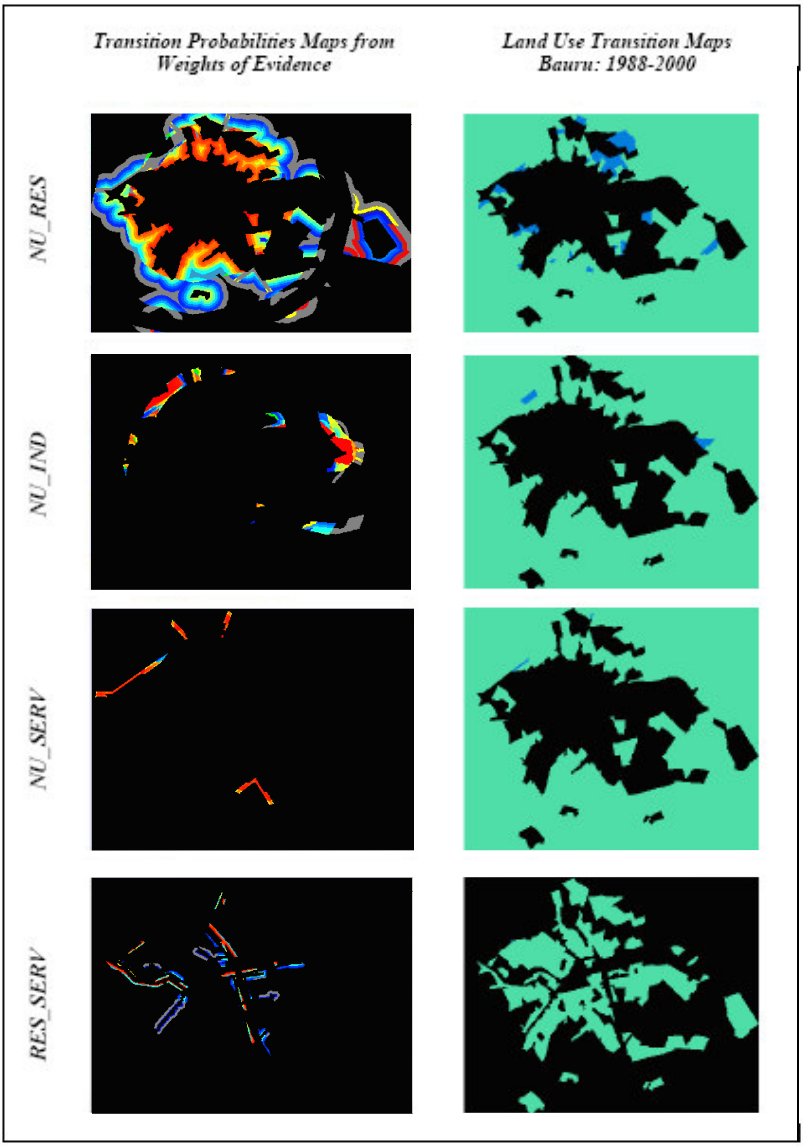

Figure 5: Estimated transition probability surfaces and land use change: 1988-2000

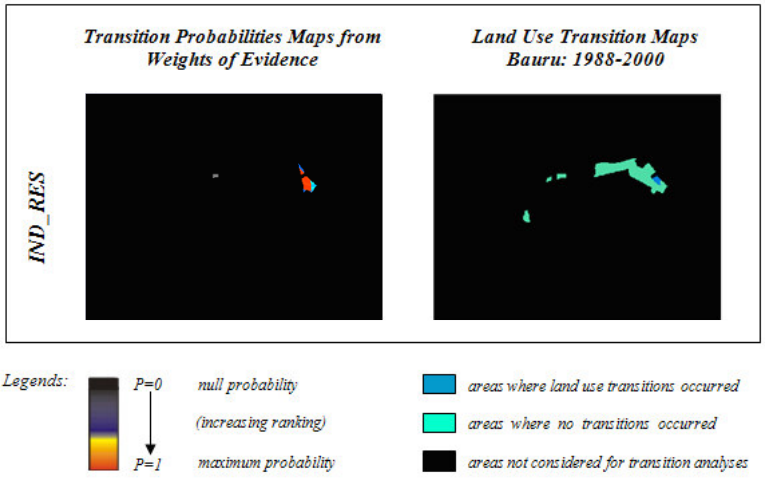

Figure 5 (Cont.): Estimated transition probability surfaces and land use change: 1988-2000

Final Remarks

Although there is a criticism towards genetic algorithms in the sense that they require manifold parameters, the GA tool of Dinamica EGO already provides the modeler with default input parameters, which have been previously tested and shown to be optimal. Genetic algorithms must be regarded as a heuristic to find an ideal solution for a problem, conducted by parallel research and not by an exhaustive and troublesome process of trial and error.

Acknowledgments

The authors wish to thank the Bauru Water Supply and Waste Water Disposal Department and the Bauru Planning Secretariat for providing the city maps. This study has received financial support from the Remote Sensing Division of the Brazilian National Institute for Space Research (DSR/INPE) and the São Paulo State Foundation for Research Support (FAPESP).

References

[1] Fogel, L. J., A. J. Owens, M. J. Walsh (1966), Artificial Intelligence through Simulated Evolution, John Wiley, New York

[2] Holland, J. H. (1975), Adaptation in natural and artificial systems, Ann Arbor, University of Michigan Press

[3] Rechenberg, I. (1973), Evolutionstrategie: Optimierung Technisher Systeme nach Prinzipien des Biologischen Evolution, Stuttgart, Fromman-Hozlboog Verlag

[4] Koza, J. R. (1992), Genetic Programming, Boston, MIT Press

[5] Jamshidi, M. (2003), Tools for intelligent control: fuzzy controllers, neural networks and genetic algorithms, Philosophical transactions, Series A, Mathematical, physical, and engineering sciences, v. 361, n. 1809, pp. 1781–808

[6] Wu, N., Silva, E. A. (2010), Artificial Intelligence Solutions for Urban Land Dynamics: A Review, Journal of Planning Literature, v. 24, n. 3, pp. 246-265

[7] Papini, L., G. A. Rabino, A. Colonna, V. Di Stefano, S. Lombardo (1998), Adaptation Learning cellular automata in a real world: The case study of the Rome metropolitan area in natural and artificial systems. In Proceedings of the Third Conference on Cellular Automata for Research And Industry, - ACRI'96, Milan, Italy, October 16-18, Springer-Verlag, pp. 165-183

[8] Goldstein, N. C. (2005), Brains versus brawn - Comparative strategies for the calibration of a cellular automata-based urban growth model. In P. M. Atkinson (Ed.), GeoDynamics, pp. 249-272, CRC Press, Boca Raton, FL, USA

[9] Al-Kheder, S., J. Wang, J. Shan (2007), Cellular automata urban growth model calibration with genetic algorithms. In Proceedings of the Urban Remote Sensing Joint Event, Paris, France, April 11-13, Kansas City, Institute of Electrical and Electronics Engineers (IEEE)

[10] Li, X., Q. S. Yang, et al. (2007), Genetic algorithms for determining the parameters of cellular automata in urban simulation, Science in China Series D-Earth Sciences, v. 50, n. 12, pp. 1857-1866

[11] Jenerette, G. D., J. G. Wu (2001), Analysis and simulation of land use change in the central Arizona-Phoenix region, USA, Landscape Ecology, v. 16, n. 7, pp. 611-626

[12] Xu, X., J. Zhang, X. Zhou (2006), Integrating GIS, cellular automata and genetic algorithm in urban spatial optimization: A case study of Lanzhou, Geoinformatics 2006: Geospatial Information Science, v. 6420, pp. 1-10

[13] Liu, Y., Y. Feng (2006), An optimised cellular automata model based on adaptive genetic algorithm for urban growth simulation. In E. Guilbert, B. Lees, Y. Leung (Eds.), Proceedings of the Joint International Conference on Theory, Data Handling and Modelling in GeoSpatial Information Science, Hong Kong, May 26-28, pp. 45-50

[14] Wu, N., Silva, E. A. (2009), Integration of Genetic Agents and Cellular Automata for Dynamic Urban Growth Modelling, In Proceedings of the 10th International Conference on GeoComputation, Sydney, Australia, November 30 – December 02, Cambridge, UK, Department of Land Economy and Darwin College of the University of Cambridge

[15] Zhang, F., (2010), Calibration of cellular automata model with adaptive genetic algorithm for the simulation of urban land-use, In Proceedings of the 18th International Conference on Geoinformatics, Beijing, China, June 18-20, Kansas City, Institute of Electrical and Electronics Engineers (IEEE)

[16] Fourotan, E., M. E. Delavar (2012), Urban growth modeling using genetic algorithms and cellular automata; A case study of Isfahan Metropolitan Area, Iran, In Proceedings of the GIS Ostrava 2012 - Surface models for geosciences, Ostrava, Czech Republic, January 23-25, Ostrava, Technical University of Ostrava

[17] Koenig, R., D. Mueller (2011), Cellular automata-based simulation of the settlement development in Vienna. In A. Salcido (Ed.), Cellular Automata – Simplicity behind Complexity, pp. 23-46, INTECH Open, Rijeka, Croatia

[18] Ashlock, D. (2006), Evolutionary Computation for Modeling and Optimization, Springer, Berlin

[19] Goldberg, D. E. (1989), Genetic algorithms in search, optimization, and machine learning, Addison-Wesley, New York

[20] Bonham-Carter, G. (1994), Geographic Information Systems for Geoscientists: Modelling with GIS, Pergamon, Ontario, Canada

[21] Hagen, A., 2003, Multi-method assessment of map similarity. In Proceedings of the 5th AGILE Conference on Geographic Information Science, Palma, Spain, April 25–27, 2002, Palma, Universitat de les Illes Balears, pp. 171–182