Reducing the search space in genetic algorithm : An application in emergence of cities

Article · June 2008

DOI: 10.1109/SMCIA.2008.5045987

K. Lavangnananda
King Mongkut's University of Technology Thonburi

Reducing the Search Space in Genetic Algorithm : An Application in Emergence of Cities

K. Lavangnananda, C. Wongwattanakarn
School of Information Technology
King Mongkut's University of Technology Thonburi (KMUTT),
126 Pra-cha-u-tid Road, Bangmod, Bangkok 10140, **Thailand**.
E-mail: kitt@sit.kmutt.ac.th , chairat@sit.kmutt.ac.th

***Abstract*— Scientists in the field of spatial economics have proposed different theories on how cities are emerged. These theories were transformed into different algorithms on emergence of cities. Assessing the efficiency of the final emergence from these algorithms is best performed where an ideal or optimal emergence is available for comparison. However, without performing exhaustive search, determination of optimal emergences from an arbitrary setup is almost impossible. This work is an application of Genetic Algorithm in determining an optimal emergence from a given setup. Ten random initial setups were generated based on Power-Law Distribution and Zipf's law. They were used in simulation of a particular emergence algorithm. Genetic Algorithm was then applied to determine optimal emergences from these setups. The very large search space in this problem prevented Genetic Algorithm from finding suitable emergences. The solution is found by using local knowledge of individuals to reduce their search space. The work affirms the benefit of Genetic Algorithm in emergence of cities and introduces a method to reduce search space in this particular area.**

I. INTRODUCTION

Emergence of cities is effected by several attributes within different facets ranging from geography to economics to environment. These variables have direct influences in the final characteristics of the emergent city [1]. There have been several studies in formation of urban cities by constructing models which allow spatial analysis to be performed. These result in several algorithms proposed on the emergence of cities. Proponent in this area is the Centre for Advanced Spatial Analysis [2]. Assessing their performance and effectiveness assumes optimal or ideal emergences are obtainable for comparison, this cannot always happen in reality. Determining optimal emergence from an initial setup is often a NP-class problem where solution cannot be expressed by formulae or algorithms. This work utilizes Genetic Algorithm (GA) in an attempt to discover an optimal emergence from a given arbitrary initial setup.

Section II introduces the field of emergence of cities and describes a particular emergence algorithm selected for this work. Models and their specification for simulation of the algorithm are explained. Results from the simulation are shown. Section III describes the application of GA in determination of an optimal emergence. Results from GA are also shown. Section IV begins with explaining the factor which hampered the performance of GA. A method to aide GA process by means of using local knowledge of individuals to reduce their search space is elaborated. Section V discusses the results and significance of the work. The paper is concluded in Section VI where future work is also suggested.

II. EMERGENCE OF CITIES

The study of the emergence of cities can be done on several aspects with different objectives. Good examples of these are stochastic cellular automata approach [3],[4], multi-based model approach [5],[6] and hybrid model approach [7]. In this work, the final formation of the city are assumed to be directly influenced by the supply and demand of individuals. The algorithm adopted in this work is of multi-agent system type proposed by C. Webster [8]. The algorithm is intentionally kept simple so that useful patterns, in terms of spatial economics, can be apparent without being influenced by too many variables.

A. The Emergence Algorithm

Major components in the algorithm which have direct influences on the formation of the city are as follows :

Number of individuals : An individual is referred as an *agent* hereafter. Each agent is both a consumer of all types and a producer of a particular type.

Type of agents : There can be many types of agents with in a community. As an agent is also a supplier of a particular type, this corresponds to different kind of producer in reality. In this paper, a notation *P* follows by subscript $_I$ is used to represent the type of producer of an agent. For examples, P_1 is used to represent a producer in which all agents need to consume on daily basis. In reality, this can mean a café or a newsagent. P_7 is used to represent a producer which are in demand on weekly basis, this can mean a petrol station, for example.

Area : In this work, the area is kept simple as a square grid.

The final emergence is directly influenced by relocation behaviour of agents. Relocation is governed by the following constraints :

Demand of consumers : As described earlier, each agent is both a consumer and a producer. Note that a P_1 agent will also has daily consumption requirement, and hence will be looking for other P_1 to satisfy the need.

Number of agents in each type of producer : This work adopts the two popular models, the Power Law Frequency Distribution [9] and the Zipf's Law [10]. Hereafter, the Power Law Distribution and the Zipf's Law models used in this work will be referred to as 'model 1' and 'model 2' for brevity.

Relocation of each agent : The assumption is made that each agent will have to satisfy the need to consume each type of produce. Therefore, for each agent, it will need to travel to a P_1 on daily basis and to P_7 on a weekly basis, and to other type of producers according to its periodical requirements. Hence, each agent will be looking to relocate itself in order to minimize the total distance it will have to travel within a period of a year. This means that relocation of all agents occur continuously and dynamically.

The algorithm can be described as follows :

Repeat
  For each agent **do**
    Repeat (find nearest producer)
      For each position in grid ordered by the distance from the agent
        If (the position isn't empty)
          And (type of producer in the position has not been found)
        Then record the position of the new producer
    Until all type of producers are found
    Find the best location for the agent such that the total distance it has to travel within 1 year is minimum
    Relocate the agent to the new best location
Until no relocation is necessary (i.e. a stable state is reached)

B. Emergence factors and specification

An analysis of the algorithm revealed that the emergence is directly influenced by the following factors :

- Total number of agents
- Total number of types of producers
- Total number of agents in each type of producer
- Grid area

Other factors which have direct influence in the simulation of the algorithm are initial setups (i.e. initial configuration of how agents are spread out initially) and computational order for relocation determination. The influence of these two factors was well studied in [11]. In this work, parameters for the emergence are types of producer (i.e. different type of P_i), grid size and number of agents in each type of producer (i.e. number of agents in each P_i). Type of producers was set to 10 and grid size was set to 100 by 100. Tables 1 and 2 summarize the number of agents in each type of producer and the total number of agents for both models.

TABLE I
NUMBER OF AGENTS IN EACH TYPE OF PRODUCERS (MODEL 1)

Type of producer	*Number*	*Type of producer*	*Number*
P_1	1024	P_{35}	32
P_7	512	P_{42}	16
P_{14}	256	P_{49}	8
P_{21}	128	P_{56}	4
P_{28}	64	P_{63}	2
Total number of agents		2,046	

TABLE II
NUMBER OF AGENTS IN EACH TYPE OF PRODUCERS (MODEL 2)

Type of producer	*Number*	*Type of producer*	*Number*
P_1	683	P_{35}	114
P_7	341	P_{42}	97
P_{14}	228	P_{49}	85
P_{21}	171	P_{56}	76
P_{28}	137	P_{63}	68
Total number of agents		2,000	

C. The simulation

The objective of the emergence is to minimize the total travel necessary for each agent over a period of time. The measure of this transaction in the final emergence in this work is taken to be the total distance in which all agents have to travel per annum. Hereafter, the value of this measure will be referred to as 'transaction value' for brevity. Even when rules of interactions among individuals are given, the science of emergences of cities cannot be formulated into equations where final outcomes can then be pre-determined. At present, modeling and simulation are the best means to study the behaviour of the emergence.

In order to ensure validity of the work, five different initial setups for each model were randomly generated and simulated using the emergence algorithm described. Figures 1 and 2 depict examples of initial setups for both models. Their final emergence are depicted in Figures 3 and 4 respectively. The characteristics of interest in the final emergence are the overall formation of clusters, their sizes and distribution within the grid area. Changes in location of individual agents are relatively less meaningful.

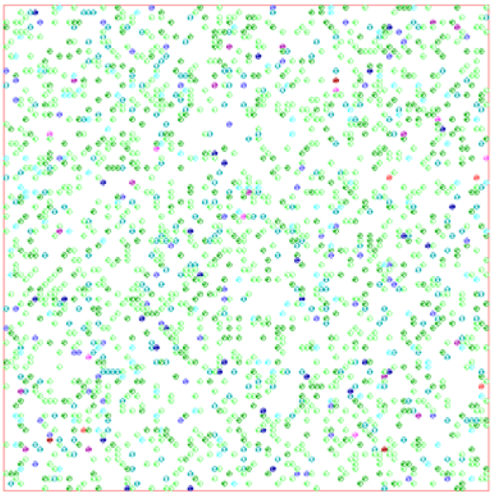

Fig. 1. An example of initial setup (model 1)

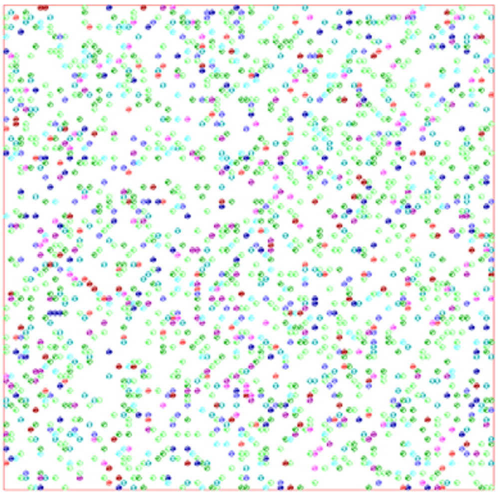

Fig. 2. An example of initial setup (model 2)

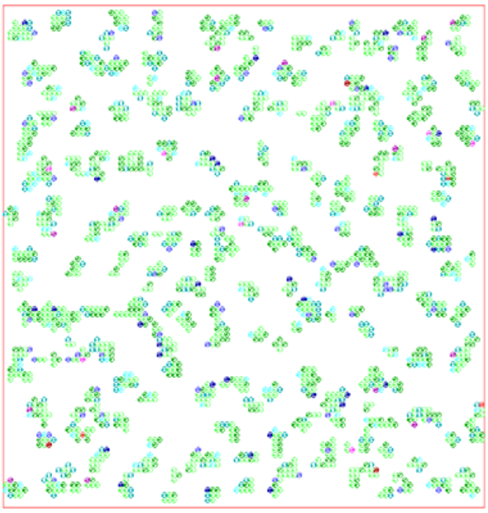

Fig. 3. An example of final emergence (model 1)

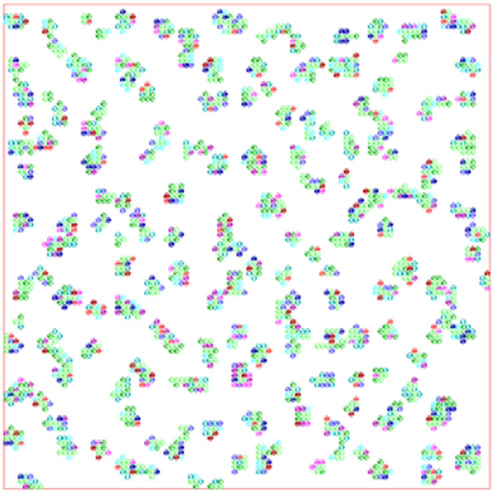

Fig. 4. An example of final emergence (model 2)

Their average transaction values (from five setups for each model) are 2,830,127.39 and 1,713,841.70 for both model 1 and model 2 respectively.

III. Application of GA in emrrgence of cities

Urban and city planning and spatial economics have benefited a great deal from advances in simulation technologies. However, application of soft computing in these fields is relatively few. Among techniques in soft computing, neural networks have the most applications, especially in managing and supporting geographical information system. Recent examples of these are [12], [13]. GA has also been applied to find a solution for multiple objective land use planning in a decision support system [14], [15].

The emergence algorithm in this work focuses on the need of each individual as the basis for emergence and optimal emergence may not be the objective. From urban planning's and spatial economics' perspective, however, having an optimal emergence from any arbitrary initial setup is very advantageous. It allows the performance of the algorithm to be assessed from several aspects, and enables also suitable and adequate urban planning.

Preliminary analysis would lead to expect that there ought to be a lot less, but larger, clusters in an optimal emergence. Without considering every possible permutation, determining an optimal emergence is almost impossible even when global knowledge (i.e. locations of all agents) is assumed. In this work, number of possible permutations in model 1 and model 2 are (10000!/8000!) and (10000!/7954!) respectively. These present very large search spaces, therefore, determination of an optimal emergence is very much an NP-class problem. Hence, evolutionary computation has an application in this field. Among several techniques existed in evolutionary computation, GA is probably the most applicable as various permutations of possible emergences can directly be represented by variety of chromosomes.

A GA system was implemented to search for an optimal emergence among vast possibilities. A chromosome represents locations of all agents within the grid area. Hence, the length of chromosomes are 2,046 (for model 1) and 2,000 (for model 2). Since number in each type of agents in chromosomes must not be altered and duplication of locations cannot be allowed. For example, although locations of some P_1 and P_{14} may be crossed, it must not result in a chromosome which does not correspond to a possible solution. Therefore, conventional GA operators (i.e. crossover and mutation) had to be modified to suit the nature of the problem.

As stated in Section II, the objective of the emergence is to minimize the transaction value (i.e. the total distance in which all agents have to travel per annum). Hence, the fitness value of each chromosome can be expressed as :

$$\text{Fitness value} = \sum_{i=1}^{N} d_i \qquad \text{……..…………..(1)}$$

where d_i is the distance in which agent i has to travel within an annum.

N is the total number of agents

The same five sets of random initial setups in each model were selected as the case study. The system was executed using PC with Pentium IV, 3.0 GHz processor. It took around an hour to complete 100 to 150 generations. After numerous experiments, suitable parametric values for GA operators are as shown in Table III.

TABLE III
SUITABLE GA PARAMETERS

Number of population	50
Selection method	Stochastic Tournament Selection
Crossover rate	0.7
Mutation rate	0.3
Termination criterion	Until performance is saturated (over 7,000 generations)

Figures 5 and 6 depict examples of the emergences for model 1 and model 2 respectively.

Their average transaction values (from five setups for each model) are 2,860,161.89 and 2,396,045.97 for both model 1 and model 2 respectively. It was apparent that GA could not yield anywhere near satisfactory emergence and did not manage to perform better than the emergence algorithm. Casual inspection of emergences suggested that its performance was probably similar to randomizing.

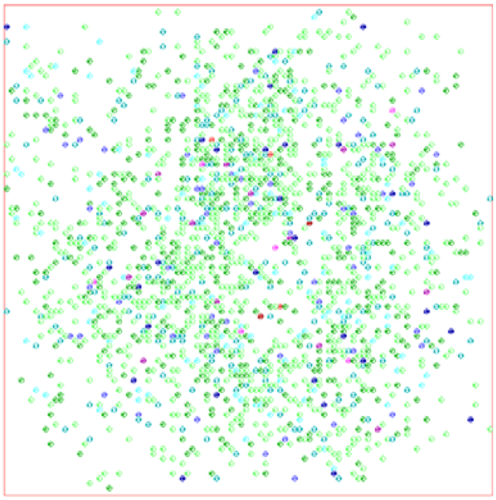

Fig. 5. An example of emergence using GA (model 1)

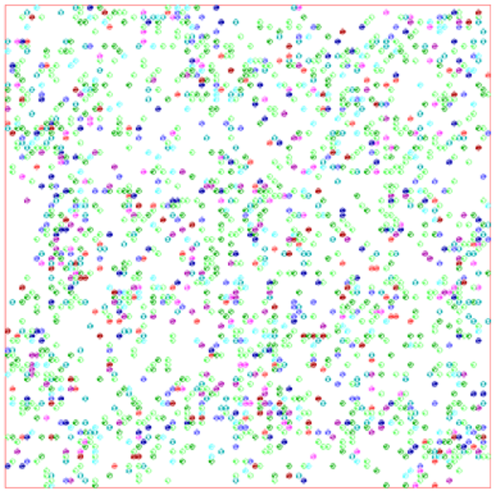

Fig. 6. An example of emergence using GA (model 2)

IV. REDUCTION OF SEARCH SPACE

The major factor which prevented GA from finding a suitable solution is the very large search space in both models (10000!/8000!) and (10000!/7954!)). Hence, effectiveness for GA operators were severely hampered and better solutions were left to chance alone. In order to aide the GA process, search space must be reduced. This was done by means of using local knowledge of each agent to limit the search space for that particular agent.

In the emergence algorithm, a new location for an agent is assigned to be at the position where distances to all type of suppliers are minimized. In other words, a polygon is established where nearest suppliers form corners of the polygon and the agent is relocated somewhere near the center of the polygon such that the transaction value is minimized. This local knowledge suggests that better locations for a particular agent ought to be within this polygon. The GA system was modified so that the search space for each agent was limited within the polygon described as above. Using Figure 7 to illustrate a determination of the search space polygon for an agent, assuming that there are only five types of agents, P_1, P_2, P_3, P_4 and P_5. In order to establish the search space polygon for the agent P_5 in the centre, the nearest P_1, to P_5 are located. A polygon is then formed by using locations of these nearest agents as its corners. Hence, a new suitable location for this particular P_5 is restricted within this search space polygon. The same concept is applied to reduce the search space for all agents in the GA process.

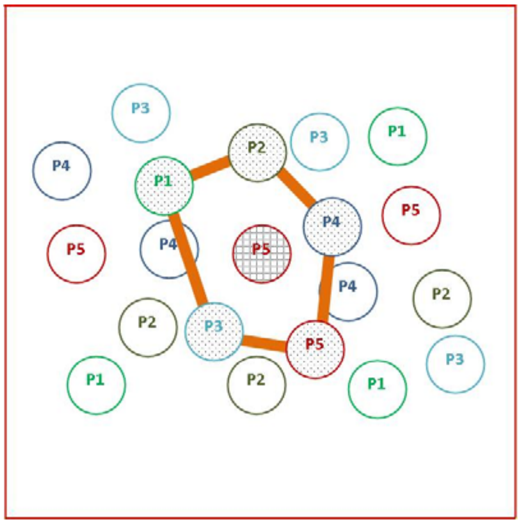

Fig. 7. An example of determining a search space polygon

The concept of limiting the search within search space polygon was applied when mutation and reproduction occurred. This reduction of search space by means of search space polygon also evolve as GA progresses through successive generations. The new GA system with reduced search space capability was applied to the same five sets of random initial setups in each model. Suitable values for GA operators were found to be the same as before, with an exception that the performance saturated at around 2,000 generations for both models. Figures 8 and 9 depict examples of the emergence from the GA with reduced search space capability.

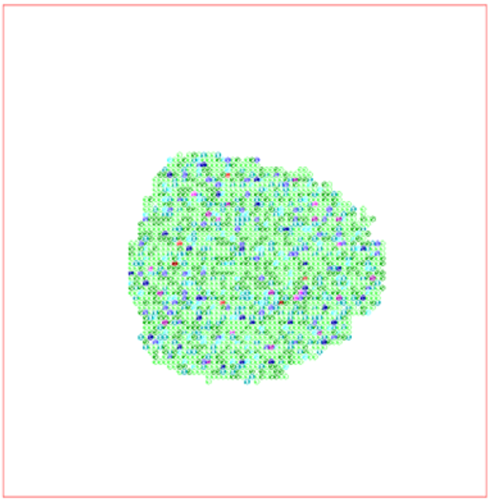
Fig. 8. An example of emergence using reduced search space GA (model 1)

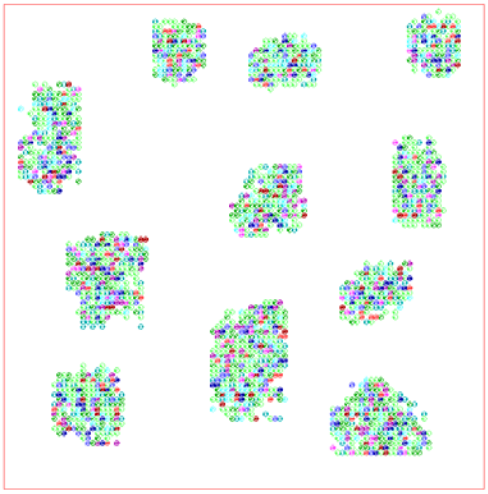
Fig. 9. An example of emergence using reduced search space GA (model 2)

Their average transaction values (from five setups for each model) are 1,712,702.15 and 1,451,306.15 for both model 1 and model 2 respectively. These values represented significantly better emergence than those from the emergence algorithm. A careful inspection also suggested that better formation of clusters too.

V. Discussion

One of the strengths of GA is the ability to produce acceptable solutions where exhaustive search is infeasible. Nevertheless, the search space in the GA process has to be manageable. The result of this work suggests that GA cannot be expected to yield acceptable performance in applications where search space is unreasonably large. Local knowledge of each individual is a mean to reduce its search space and aide the GA process. The power of search space reduction is very crucial in this work. Figures 5 and 6 indicate that apart from difficulty in formation of clusters, the significance of different proportions between model 1 and model 2 had very little effect. On the contrary, Figures 8 and 9 reveal that reduction of search space allowed favourable formation of clusters and also the effect of different proportions in both models to emerge.

Although GA with reduced search space cannot guarantee optimal emergence, the results may be used as a model to assess an emergence algorithm (not limited to the one described in this work). While the objective of the emergence algorithm is not to produce an optimal emergence, comparison between their emergences can be done from several aspects. Comparison and analysis of this nature is beyond the scope of this paper and lies within the field of spatial economics. However, the most apparent feature from the emergence is that proportion of each type of producer in the total population (i.e. number of agents in each type of producer) has a strong influence in the final emergence and formation of clusters. Optimal emergence, therefore, depends a great deal on these proportions and especially on the number of agents which belong to lesser demand suppliers.

VI. CONCLUSION AND FUTURE WORK

Evolutionary computation, and GA in particular, has an application in the fields of urban planning and spatial economics. This work introduces yet another application of GA in emergence of cities. It stresses that search space in GA must not be unreasonable in order to find acceptable solutions. In situations where search space is initially too vast, local knowledge of individuals can reduce this to a manageable level.

Future work can be carried out in several facets. The technique of reducing search space in GA in this work can be experimented on other emergences similar to the one in this work. This can lead to more approaches to reduction of search space which may have wider implication. As mentioned in Section III, application of GA in this work is very resource demanding in terms of computation. This is a suitable candidate for parallel processing approach to GA [16]. A recent development in GA known as *cellular GA* [17] where search space for individuals in GA are based on their neighborhood may also be investigated and adapted in order to the existing system.

Acknowledgment

The authors wish to express their gratitude to Prof. Dr. C. Webster for initiating the work and invaluable input on emergence of cities. The support of computing facilities at School of Information Technology (SIT) at King Mongkut's University of Technology (KMUTT), Thailand is also gratefully acknowledged.

References

[1] S. E. Page, "On the Emergence of Cities", *Journal of Urban Economics*, vol. 45, pp. 184–208, 1999.

[2] Centre for Advanced Spatial Analysis, UCL, http://www.casa.ucl.ac.uk/research/ c_a.htm.

[3] M. Batty, H. Couclelis, and M. Eichen, "Urban systems as cellular automata", *Environment and Planning B-Planning & Design, Vol. 24, Issue 2*, pp., 159–164, 1997.

[4] I. Muliata and Y. Hariadi, "Urban Area Development in Stochastic Cellular Automata", *Urban/Regional 0412001*, EconWPA, http://129.3.20.41/eps/urb/papers /0412/0412001.pdf, 2004.

[5] K. Chen and E. G. Irwin, C. Jayaprakash, and K. Warren, "The Emergence of Racial Segregation in an Agent-Based Model of Residential Location: The Role of Competing Preferences**",** *Journal Computational & Mathematical Organization Theory, Vol. 11*, pp. 333-338, 2005.

[6] P. Waddell, "Reconciling Household Residential Location Choices and Neighborhood Dynamics", Under revision, *Sociological Methods and Researc*h, 2006.

[7] P. M. Torrens and D. O'Sullivan, "Cellular automata and urban simulation: where do we go from here?", *Environment and Planning B, Vol. 28*, pp. 163–168, 2001.

[8] C. J. Webster, "The New Institutional Economics and the evolution of modern urban planning: Insights, issues and lessons", *Town Planning Review, Vol. 76, Issue 4*, pp. 471-500, 2005.

[9] C. J. Webster, and L. W .C. Lai, *Property Rights, Planning and Markets: managing spontaneous cities,* Cheltenham UK, Edward Elgar ISBN 1-84376-341-9, 2003.

[10] Y. Mansury and L. Gulyás, "The emergence of Zipf's Law in a system of cities: An agent-based simulation approach", *Journal of Economic Dynamics and Control, Vol. 37, Issue 7*, pp. 2438-3460, 2007.

[11] K. Lavangnananda and C. Wongwattanakarn, "The Effects of Different Initial Setups and Computation on Simulation of Emergence of Cities", *Proc. of the 10th Int. Conf. on Computers in Urban Planning and Management (CUPUM'07)*, 11-13 July, Iguassu Falls, Brazil, 2007.

[12] L. Diappi, et. al., "The driving forces of urban gentrification : An investigation on centrifyers profile through neural networks", *Proc. of the 10th Int. Conf. on Computers in Urban Planning and Management (CUPUM'07)*, 11-13 July, Iguassu Falls, Brazil, 2007.

[13] F. Lúcio Zampieri and D. Rigatt "The use of artificial neural networks on the creation and implementation of a predictive model of *the pedestrains*", *Proc. of the 10th Int. Conf. on Computers in Urban Planning and Management (CUPUM'07)*, 11-13 July, Iguassu Falls, Brazil, 2007.

[14] R. J. Balling, el. al., "Multiobjective Urban Planning Using Genetic Algorithms", *Journal of Urban Planning and Development, Vol. 125, No. 2*, pp. 86-99, 1999.

[15] T. J. Stewart, R. Janssen and M. Van Herwijnen, "A genetic algorithm approach to multiobjective landuse planning", *Computers & Operations Research, Vol.31, Issue 14*, pp. 2293-2313, 2004.

[16] *S.* A Massively Distributed Parallel Genetic Algorithm (mdpGA), *Tech. Report CMUCS -92-196, Carnegie Mellon University, School of Computer Science, Oct.13, 1992.*

[17] E. Alba and B. Dorronsoro, *Cellular Genetic Algorithms*, Operations Research/Computer Science Interfaces Series, Springer-Verlag 2007.