

An Architecture Based on Multi-Agent Systems and Geographic Databases for the Development of Georeferenced Ecological and Social Simulations

Pablo Souza Grigoletti^{1,*}, Antônio Carlos da Rocha Costa^{1,2}

¹Instituto de Informática/PPGC – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

²Escola de Informática/PPGINF – Universidade Católica de Pelotas (UCPel)
Rua Félix da Cunha, 412 – 96.010-000 – Pelotas – RS – Brazil

grigoletti@gmail.com, rocha@atlas.ucpel.tche.br

Abstract. *This work is situated in the intersection of 4 different areas: Social Simulations, Ecological Simulations, Multi-Agent Systems and Geocomputing. Its main objective is to propose a multi-agent architecture designed for the development and execution of geographically referenced social and ecological simulations, that is, social and ecological simulations that use geographically referenced data taken from a Geographic Database, supporting the dynamic modeling of social and/or ecological systems. First of all, is introduced the motivations and objectives of the development of this new architecture, detailing the advantages of join these 4 areas, and the needs of the existent architectures. The features of this architecture and a complete study case are presented too. Finally, a comparison with others architectures and the conclusions are described.*

1. Introduction

Studies about the relationships between the mankind and its environment are each time more present in the scientific community. In this way, the Computer Science has been giving important contributions, providing simulation tools for making the analysis of this kind of relationship. A great number of computer models have been developed to study the complex systems that compose these relationships.

A model is a simplification – smaller, with less details and less complexity – of a structure or system [Gilbert and Troitzsch 1999]. In general, the process of creating a model is one of the most important pieces of the simulation development.

Simulations are very helpful when it is necessary to understand complex systems. In many cases, researchers do not have success when they model systems using pure analytical mathematical methods. Simulations of social and ecological phenomena may then be used as tools for analyzing and understanding the complexity of the phenomena.

For instance, when studying the consequences that human actions produce on the environment, those simulations focus on modeling the set of effects of the population of agents on the common environment, by the study of aspects such as the complexity,

*CNPq scholarship holder - Brazil.

emergence, self-organization and dynamics created by the agents' actions on the environment [Castelfranchi 1998][Ferber 1999].

The use of Multi-Agent Systems (MAS) as a tool for the development of social and ecological simulations strengthens the scientific studies in those areas. The main reason for this is that it is possible to directly apply the concepts of agents and agents' society on the modeling of natural societies. In this way, nowadays MAS are frequently used for this kind of simulation [Gilbert and Troitzsch 1999].

On the other hand, the study of spatial phenomena resulted, in the last decades, in the development of specific computational systems for the modeling and analysis of the geographic space: the Geographic Information Systems (GIS). A GIS is an information system that allows the capture, modeling, manipulation, recovery, analysis and representation of geographically referenced data [Worboys 1995]. In a GIS, the storage component generally is called Geographic Database (GDB).

The use of georeferenced social and ecological data (stored in a GDB, for example) helps a lot to increase the realism of the simulation of anthropic actions, that is, the men's actions on the environment. It allows to show clearly the dependencies that may exist between the social process and the physical environment where it happens. This kind of social and ecological simulation is frequently called geographically referenced social simulation [Boero 2006].

This work presents an architecture based on MAS that supports the development and execution of geographically referenced social and ecological simulations. Simulations performed using this architecture may access spatial information available in a GDB.

The paper is organized as follows. In Section 2, the main motivations and objectives of the work are introduced. A review of related works is presented in Section 3. Section 4 details the MAS architecture for georeferenced social and ecological simulations. A complete study case is presented in Section 5. Section 6 brings the conclusion.

2. Motivations and Objectives

The four main motivations for this work are:

- according to [Bordini et al. 2005], in the MAS area, and consequently in social simulations based on MAS, the modeling and representation of the environment where the agents are situated is an extremely important aspect, however not so much explored. In reactive agent systems, the agents do not have memory and, in this case, the environment has an important function, since it is only by perceiving the environment that agents can take decisions. In cognitive agent systems, the agents have an internal representation of the environment in which they are situated. In this case, these agents take their decisions based on the changes that their perceptions cause in their internal representation. In both cases, the modeling of the environment is an extremely important activity in the of the simulations.
- for some time now, Cellular Automata (CA) have been used in simulation models to represent social and ecological environments. However, the traditional CA theoretical base [Neumann 1966] does not allow that the automata move in the environment, the only changes allowed are the changes in the information in the

automata cells. In a MAS based model both are possible: information concerning locations in the environment can be changed, and the automata (in this case, agents) can move in the environment;

- Traditionally, Geocomputing emphasized the representation of spatial phenomena in a static way, so that the main abstraction used is the map. However, there are a lot of spatial phenomena that are dynamic, and this static representation can not capture it well. In this way, abstractions that allow an appropriate representation of spatio-temporal phenomena are a need in this area [Pedrosa and Câmara 2004], and recent work has been done to supply this need (e.g. [Rocha et al. 2001]);
- strengthening the need to develop spatio-temporal models, [Santos 1996] has described the space as “indivisible of the human beings that inhabit and modify it all the time”.

Based on these motivations, the main objective of this work is to provide an architecture, based on MAS and GDB concepts, for the development and execution of georeferenced social and ecological simulations. We intend to analyze the advantages of joining these two areas in the simulation context, focusing on the spatio-temporal representation of the entities that exist in a physical environment.

3. Related Works

With the aim of identifying the important features for a new simulation architecture, an analysis of the features of some agent-based simulation platforms was done: (i) *Swarm* [Minar et al. 1996], (ii) *Repast* [North et al. 2006], (iii) *SeSam* [Klügl et al. 2006], (iv) *NetLogo* [Tisue and Wilensky 2004], (v) *OBEUS* [Torrens and Benenson 2005] e (vi) *SMA-SIG* [Gonçalves 2003]. The main results of this analysis are:

- few platforms provide abstract functions to create movement behaviours in the modeled entities; however, this kind of behaviour is very important to obtain more realistic simulations.
- a dynamic connection with the GDB allows a better use of the geographic data, providing an easy way to do complex spatial queries; however, few platforms provide such feature.
- it is not enough to have simulation platforms that allow the modeling of the environment and of the geographic attributes of entities; it is necessary to create perceptions and behaviours for the agents that use these features.
- it is important to use programming structures that allow users with little programming expertise do create simulation models without having to expend large efforts to learn programming techniques;
- it is necessary that the platform provides some well accepted standard for the communication between the active entities of the simulation model. These communication means are useful when it is necessary to create simulations (situations) that the entities should cooperate;
- the use of a discrete-time scheduler makes easy the simulations analysis, allowing the user to check simulation information at each time step.

In this way, the proposed architecture intend that puts together the positive features of the available simulation platforms, and overcome the shortcomings that they present. This architecture and its features are presented in the next section.

4. Proposed Architecture

In the proposed architecture there are two basic kinds of agents: **mobile agents** (agents that can change their position on the environment) and **fixed agents** (agents that can not change their position on the environment). Both mobile agents and fixed agents occupy space in the environment. These agents are specialized according to their geometric shape: *point agent*, *line agent*, *polygon agent*. These agents' shapes have a direct relationship with the ones used in GDB to represent objects from the real world. There is also a special kind of agent that does not have shape and does not occupy in the environment, and may be used to gather information about the simulation in a way programmed by the user (in addition to the automatic gathering of information performed by the simulator).

In this architecture the agents are organized using the concept of *layer of agents*. This organization mode makes easy the development of a model and does not interfere in the ways the agents interact. In an abstract vision of the architecture (presented in Figure 1), it is possible to identify three main layers:

- **auxiliar layer:** it is an optional layer. It contains auxiliar agents. The use of these agents aims in help on the development of the model. They are used to collect some information to ease the analysis of the simulations and do not represent agents that exist in the system being simulated;
- **social layer:** it contains the agents of the society that is being modelled;
- **spatial layer:** it contains the agents of the environment that is being represented.

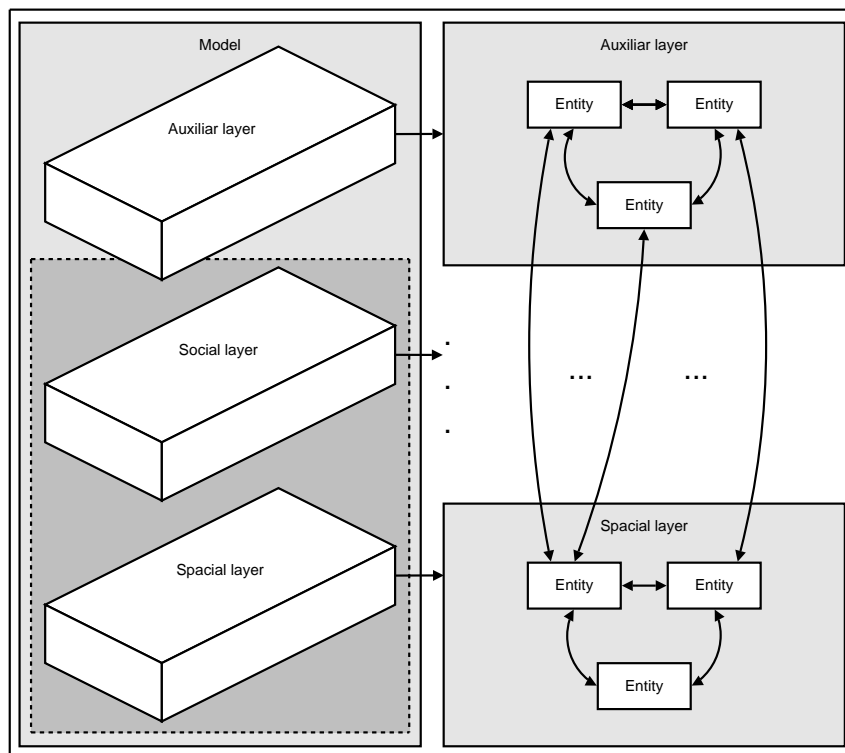


Figure 1. Abstract model of the proposed architecture.

In some cases, the social and spatial layers can be coupled and represented as just one layer. This unique representation strengthens the ideas presented in [Santos 1996],

where the geographic space is described as an indivisible of the human beings that inhabit and modify it all the time. In such cases, the aim is to join the elements that compose the environment (the geographic objects that represent the real world) and the events that change the structure of this environment (the human actions and physical processes).

When the agents are created, their shape, location, attributes, perception and behaviour is defined. In general, the location and some other attributes are data from the GDB. The shape, behaviour, perception and another attributes are defined by the user (possibly in a configuration file), as is illustrated in Figure 2. In this way, *the agents in the simulation model represent, in a form subject to dynamic evolution, the static data gathered from the GDB.*

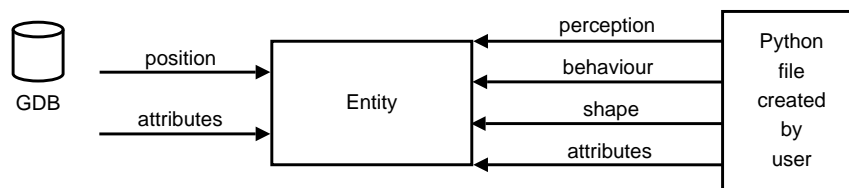


Figure 2. Definition of an agent.

The scheduling of the agents' behaviours is based on a discrete event system scheduler. It operates according to two possible policies: sequential or random. Given the problems that the scheduling policies may introduce [Michel et al. 2001], both policies are necessary to make sure that the scheduler will not influence the simulation results.

The agents' behaviours are defined using the Python language syntax, which is a "Very High Level Language" (VHLL) and has a very simple syntax.

Mobile agents may change their position in the environment and then can explore different places. In this way, mobile agents can perform different actions, according to the places in which they are. To achieve more realistic simulations in an easier way, the architecture provides a set of pre-defined high-level abstraction movement behaviours, such as the ones presented in [Reynolds 1999], which the user/programmer may easily incorporate in his agents.

The integration with the GDB happens in a dynamic way. The access to the geographic data is realized during all the simulation period. In this way, all the GDB functions and operators can be used by the entities. In this context, some important tools, techniques and structures of Geocomputing may be directly used in the simulations, allowing the use of more detailed geographic information and consequently the development of better spatial models for simulations.

Two or more agents can be declared to be adjacent. When this happens and one of the agents changes its shape or position, the shape and position of the adjacent agents are modified too. The main motivation to introduce this feature is that it makes possible to use agents with adjacent borders, just like territorial divisions of quarters and cities, the modeling of edges of lakes, the division of countryside areas etc., that behave so that when border is changed the adjacent borders are changed too.

Also, a number of perception processes for the agents (based on distance, kind of perceived agent and kind of layer) are pre-defined in the architecture. Moreover, the com-

munication between the agents can occur either in a direct form (by message exchanges), or in an indirect form (using a shared data structure, called a *blackboard*).

A prototype of the architecture, called **GeoReferenced Simulations Platform (GRSP)**, was developed. The main objective of this development was to provide a better way to evaluate all the features and functionalities of this architecture. The main computational tools used for that were: **(i) Python programming language**, **(ii) PostgreSQL database system**, **(iii) PostGIS extension and GEOS library** (used to allow that PostgreSQL uses vectorial geographic data, according to the OGC-SFS standards [Open Geospatial Consortium 2007]). The organization of the prototype system is illustrated in Figure 3.

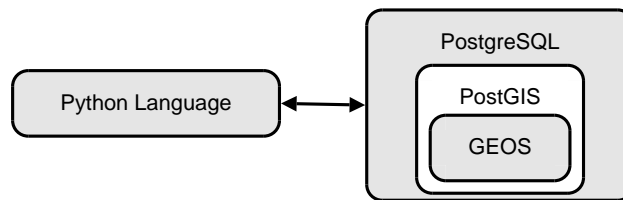


Figure 3. Organization of the main used computational tools.

Figure 4 presents the graphical user interface of the prototype system. A number of social and ecological simulations were developed on this prototype. One of them is presented as a case study, in Section 5.

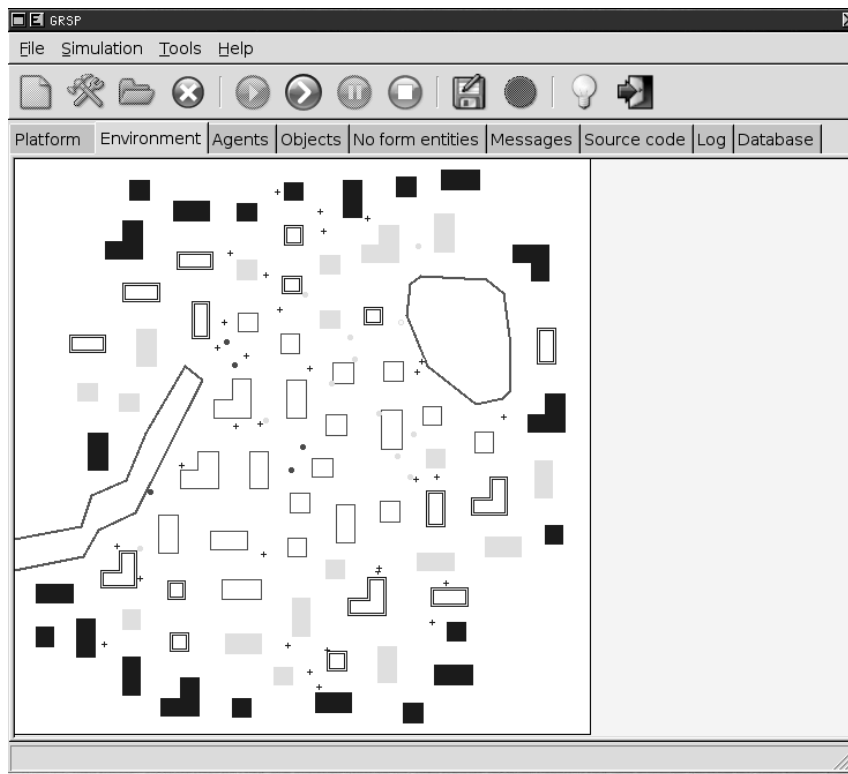


Figure 4. The GRSP graphical interface allows the control of the simulations, the visualization of entities information, their positions, as well as the Python file, log file and GDB information.

5. Study Case

This study case is inspired on the fusion of **some ideas** of two study cases, both presented previously in [Grigoletti 2007]. The first one is about the peripherisation process [Barros and Sobreira 2002], which is characteristic of Third World cities, more specifically of Latin American cities. Peripherisation can be defined as a kind of growth process characterised by the expansion of borders of the city through the formation of peripheral settlements, which are, in most cases, low-income residential areas. The second one is about the influence of policemen actions on crime number, in a certain urban area [Vasconcelos and Furtado 2005]. These two ideias are just used as motivation, in order to create the following social simulation study case.

5.1. The Scene

In this scene, the population is divided in three distinct economic groups according to the pyramidal model of distribution of income in Third World countries. All population have the same locational preferences, that is, they all want to inhabit close to the areas that are served by infrastructure, with nearby commerce, job opportunities, security (police stations), and so on. As in Third World cities these facilities are found mostly close to the high-income residential areas, the preference of location is to be close to a high-income group residential area.

What differentiates the behaviour of the three income groups is the restrictions imposed by their economic power. Thus, (i) the high-income group is able to inhabit in any place of its preference; (ii) the medium-income group can inhabit everywhere except where the high-income group is already inhabiting and, in turn, (iii) the low-income group can inhabit only in the vacant space.

On the other hand, in this scene there are policeman and criminal groups. When a policeman is near a criminal, usually he will capture it and put it in the jail (penitentiary). If an area usually has more criminals than policemen, this area is considered an area with a high crime rate. High-income groups only inhabit areas with a low crime rate. Medium-income groups inhabit areas with low or medium tax of crime. Low-income groups do not mind about that.

The main objective of this study case is to discover the relationship between the amount of crimes in a certain urban area and the economic profile of the group that locates in this area.

5.2. The Model

To create the spatial model was used the urban area of Porto Alegre city (Brazil), represented by four maps, shown in Figure 5 (all the maps were stored as vectorial data in a GDB):

- **urban area map:** this map contains information about the streets and blocks of the studied area;
- **real estate map:** this map contain information about the real estate that can be acquired by people to build/buy/lend the buildings;
- **police stations map:** this map contain information about the police stations;
- **penitentiary map:** this map contain information about the penitentiary.

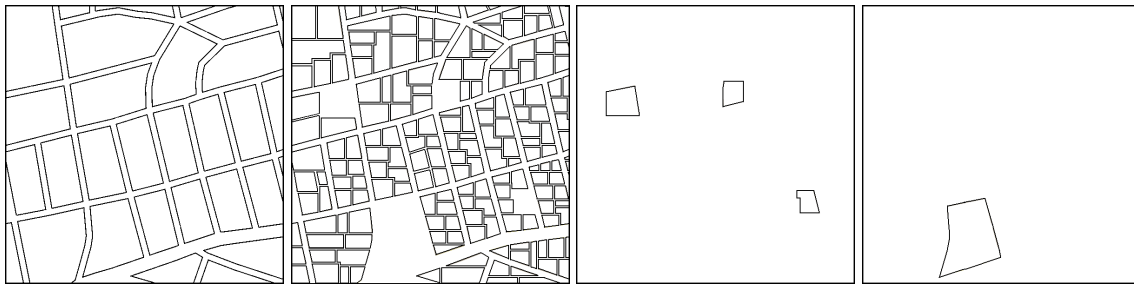


Figure 5. In order, from left to right, the urban area map, the real estate map, the police stations map, the penitentiary map.

In this model were created ten kinds of agents:

- **policeman agents (*mobile point agents*):** these agents are created in some police station. They move around the urban area (avoiding obstacles) looking for criminals. When they find some criminal agent they capture it and put it in the jail (penitentiary);
- **criminal agents (*mobile point agents*):** these agents are created in random places in the urban area. They move around the urban area (avoiding obstacles);
- **high-income, medium-income, low-income agents (*mobile point agents*):** these agents are created in random places in the urban area. They move around (avoiding obstacles) looking for a real estate (to build/buy/rent a building);
- **block agents (*fixed polygon agents*):** these agents contain the real estates of the urban area;
- **real estate agents (*fixed polygon agents*):** these agents represent the real estates of the urban area (that will be bought/etc.);
- **police station agents (*fixed polygon agents*):** these agents represent the police stations of the urban area;
- **penitentiary agent (*fixed polygon agent*):** this agent represents the penitentiary;
- **auxiliary agents:** these agents are used to get information about the simulation.

5.3. Results

The policeman agents are represented by black dots (●) and the criminal agents are represented by white dots (○). The real estates are bought by high-income agents are in light gray, the ones bought by medium-income agents are in dark gray and the ones bought by low-income agents are in black. Figures 6, 7, 8 represent some steps of this simulation. High-income, medium-income, low-income agents were not pictured, to allow for clearer images.

6. Conclusion

The proposed architecture seems to bring the following contributions:

- **to the Social Simulation area:** the architecture of a simulation system (and a prototype) that allows for a detailed spatial modeling of (social) environmental entities, based on Geocomputing data structures, more specifically vectorial data from a GDB;

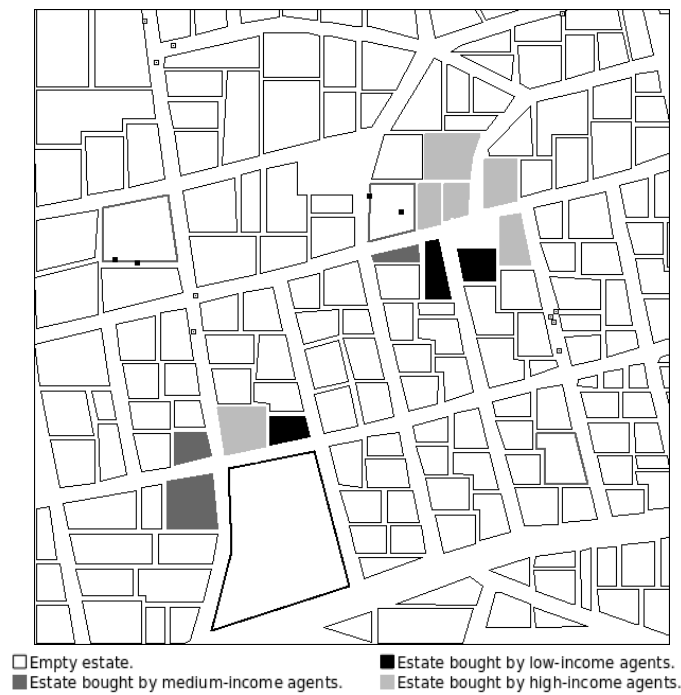


Figure 6. The simulation in its 13th step. The simulation is in the beginning. Few agents have bought a real estate. Few policeman and criminal agents are on the streets.

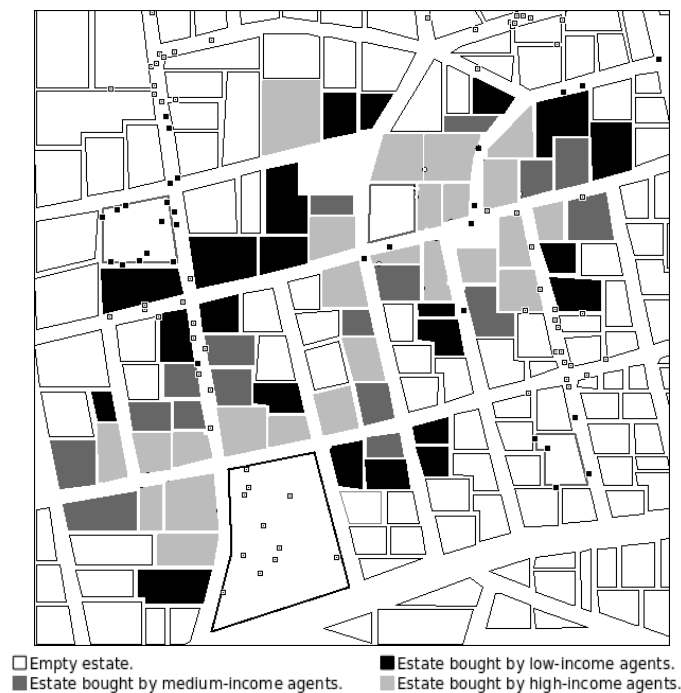


Figure 7. The simulation in its 80th step. It is possible to see that the highest concentration of real estates bought by high-income agents is near a police station, because they are the agents that have higher chances to succeed in buying such areas.

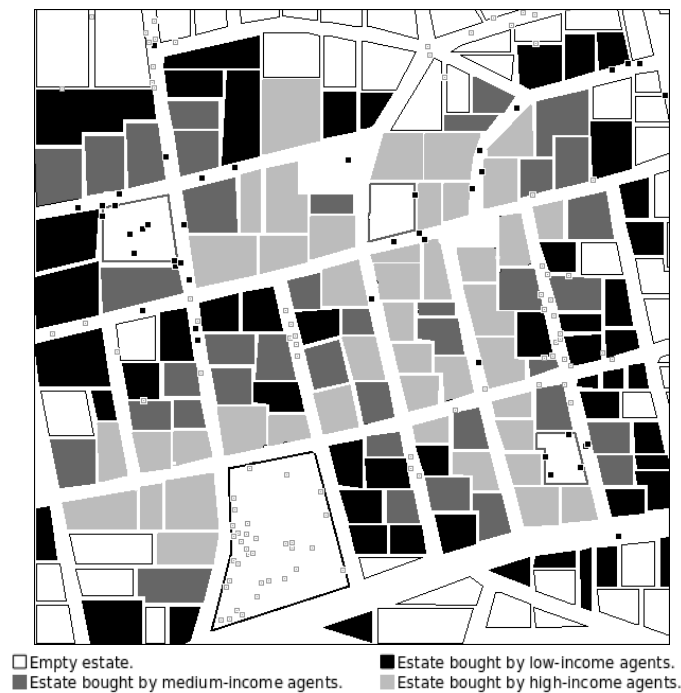


Figure 8. The simulation in its 155th step. It is clear the peripherisation process in this simulation. In the center of the area, high-income real estates (light gray), around it medium-income real estates (dark gray) and near the borders of this area are the low-income real estates (black). There are some center points with low-income real estates because in these areas there are a lot criminals and few policeman agents.

- **to the Georeferenced Simulation area:** the architecture of a simulation system (and a prototype) that allows for the development of detailed modeling of anthropic processes, modeling not just actions of isolated individuals, but also the effects of social interactions on those actions, given that the simulation of direct interactions between agents is possible;
- **to the Geocomputing area:** an abstraction tool to represent dynamic spatio-temporal processes and events, using MAS based simulations.

The computational tool GRSP can be considered another contribution of this work. The study case was presented just to show the use of some developed features. Beyond it, the architecture contribute in the way it joins some needs of others architectures.

The final contribution can be described as: *to provide the structure that are needed to allow for the use of MAS in social and ecological simulations, using geographic models created by GDB and so that MAS can generate spatial data to be used by GDB.* In other words, the work seems to have shown that both Geocomputing and MAS based simulations can benefit from the MAS-GDB coupling.

A comparison of the features found in the architectures studied in Section 3 with the features of the proposed architecture (and created on GRSP software) is presented in Table 1.

Table 1. Comparative table of the MAS based simulations architectures.
Legend: ✓: *feature implemented*, *: *feature partially implemented*, blank: *feature not found or not implemented*.

Feature	Proposed Architecture	Swarm	Repast	SeSAM	NetLogo	OBEUS
High level movement behaviours	✓					
Use of raster data		*	✓		✓	
Dynamic connection with GDB	✓					
Use of vector data	✓					✓
Easy way to develop	*		*	✓	*	
Communication based on FIPA standard	✓			✓		
Discrete time scheduler	✓	✓	✓	✓	✓	✓
Auxiliar entities	✓	✓				
Adjacent entities	✓				✓	

References

- Barros, J. and Sobreira, F. (2002). City of Slums: self-organisation across scales. *CASA Working Paper Series*, (55).
- Boero, R. (2006). The spatial dimension and social simulations: A review of three books. *Journal of Artificial Societies and Social Simulation*, 9(4).
- Bordini, R. H., Costa, A. C. d. R., Hübner, J. F., Moreira, Á. F., Okuyama, F. Y., and Vieira, R. (2005). MAS-SOC: a social simulation platform based on agent-oriented programming. *Journal of Artificial Societies and Social Simulation*, 8(3). Available at <http://jasss.soc.surrey.ac.uk/8/3/7.html>.
- Castelfranchi, C. (1998). Simulating with cognitive agents: The importance of cognitive emergence. In Sichman, J. S., Conte, R., and Gilbert, N., editors, *Proceedings of the First International Workshop on Multi-Agent Systems and Agent-Based Simulation (MABS '98)*, volume 1534 of *Lecture Notes in Computer Science*, pages 26–44, Berlin, Germany. Springer-Verlag.
- Ferber, J. (1999). *Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence*. Addison-Wesley Longman Publishing Co., Inc., Boston, USA.
- Gilbert, N. and Troitzsch, K. G. (1999). *Simulation for the Social Scientist*. Taylor & Francis, Inc., Bristol, PA, USA.
- Gonçalves, A. (2003). Multi-Agentes para Simulação em Sistemas de Informação Geográfica. Master's thesis, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Lisboa, Portugal.
- Grigoletti, P. S. (2007). Uma Arquitetura Baseada em Sistemas Multiagentes para Simulações em Geoprocessamento. Master's thesis, UFRGS, Porto Alegre, RS, Brasil.
- Klügl, F., Herrler, R., and Fehler, M. (2006). SeSAM: implementation of agent-based simulation using visual programming. In *Proceedings of the fifth international joint*

- conference on Autonomous agents and multiagent systems (AAMAS '06)*, pages 1439–1440, New York, NY, USA. ACM Press.
- Michel, F., Ferber, J., and Gutknecht, O. (2001). Generic Simulation Tools Based on MAS Organization. In *Proceedings of the 10 European Workshop on Modelling Autonomous Agents in a Multi Agent World (MAMAAW'2001)*.
- Minar, N., Burkhart, R., Langton, C., and Askenazi, M. (1996). The swarm simulation system: a toolkit for building multi-agent simulations. Working Paper 96-06-042, Santa Fe Institute, Santa Fe, New Mexico, USA.
- Neumann, J. V. (1966). *Theory of Self-Reproducing Automata*. University of Illinois Press, Champaign, IL, USA.
- North, M. J., Collier, N. T., and Vos, J. R. (2006). Experiences creating three implementations of the repast agent modeling toolkit. *ACM Transactions on Modeling and Computer Simulation*, 16(1):1–25.
- Open Geospatial Consortium (2007). OpenGIS Simple Features Specification For SQL Revision 1.1.
- Pedrosa, B. and Câmara, G. (2004). *Modelagem Dinâmica e Geoprocessamento*. EM-BRAPA, Brasília, DF, Brasil.
- Reynolds, C. W. (1999). Steering Behaviors For Autonomous Characters. In *Proceedings of Game Developers Conference*, pages 763–782, San Francisco, CA, USA. Miller Freeman Game Group.
- Rocha, L. V., Edelweiss, N., and Iochpe, C. (2001). Geoframe-T: a temporal conceptual framework for data modeling. In *Proceedings of 9th ACM International Symposium on Advances in Geographical Information Systems*, pages 47–52. ACM Press.
- Santos, M. (1996). *A Natureza do espaço: técnica e tempo, razão e emoção*. Hucitec, São Paulo, SP, Brasil.
- Tisue, S. and Wilensky, U. (2004). Netlogo: A simple environment for modeling complexity. In *Proceedings of the International Conference on Complex Systems (ICCS2004)*, pages 16–21.
- Torrens, P. M. and Benenson, I. (2005). Geographic Automata Systems. *International Journal of Geographical Information Science (IJGIS2005)*, 19(4):385–412.
- Vasconcelos, E. and Furtado, V. (2005). Um Simulador Tutorial Multi-Agente para Treinamento da Alocação de Equipes Policiais. In *Anais do XVIII Encontro Nacional de Inteligência Artificial*, pages 892–901, Porto Alegre, RS, Brasil.
- Worboys, M. F. (1995). *GIS: A Computing Perspective*. Taylor & Francis, London, England.