Agent Based Simulation for the Study of Complex Social Systems

Adolfo LÓPEZ-PAREDES[1], Juan PAVÓN[2]
[1]Universidad de Valladolid, Valladolid, Spain
[2]Universidad Complutense de Madrid, Madrid, Spain
adolfo@insisoc.org, jpavon@fdi.ucm.es

In this paper we argue how agent based simulation constitutes a new paradigm for the study and characterization of complex social systems. There is a growing interest among computer scientists in developing shells and specialized software for social scientists. We summarize the main features of some of these tools, and in particular, the usability of INGENIAS, a multi-agent system development framework, that combines graphical modeling and code generator facilities to provide the source for simulations.
Keywords: *agent-based models, simulation, complexity, INGENIAS*

1 Introduction

There is an increasing interest in Agent-based Social Simulation as a new paradigm to study Complex Systems. While the physical world is considered composed of systems that are linear or approximately linear, it is evident that human societies, economics, institutions and organizations are complex systems, using *complex* in the technical sense to mean that the behavior of the system as a whole cannot be determined by partitioning it and understanding the behavior of each of the parts separately (a classic strategy of physical sciences).

Complexity is a term used in many ways according to different schools; see [2] for a survey. Most users come from the field of non-linear dynamic models as applied to Economics. Complexity is then, the fourth C in this line of research: cybernetics, catastrophe, chaos and complexity. But the field of complexity is controversial and unsettled. And there is no accepted definition of the term. There have been two prevailing views. A dynamical system is complex if it endogenously does not tend asymptotically to a fixed point, a limit cycle, or an explosion. Alternatively a situation exhibits complexity when there is an extreme difficulty of calculating solutions to optimization problems. We adopt this view of complexity that in turn comes from two sources:

- From the aggregate outcome of simple agents' myriad interactions taking notice of each other agents' actions: institutional complexity. In these interactions, agents relate to each other and with the environment through agent–environment production rules, and agent–agent rules. We release an initial population of agents into the simulated environment and watch for macroscopic spontaneous order.
- From agents trying to model other agents modeling of them, modeling those agents, *ad infinitum*. This is the main source of complexity in game theory. Expectations about other agents' strategic behavior (which is the view in [9]).

2 Agent-Based Simulation

Simulation is a third way of doing science [1], and an important type of simulation in Social Sciences is *agent-based modeling*. This type of simulation is characterized by the existence of many agents that interact with each other with little or no central direction. The emergent properties of an agent-based model are then the result of a *bottom-up* process, rather than *top-down* direction.

A multi-agent model consists of a number of software entities, the *agents*, interacting within a virtual environment. The agents are programmed to have a degree of autonomy, to react to and to act on their environment and on other agents, and to have goals that they aim to satisfy. In such models, the agents can have a one-to-one correspondence with the individuals, organizations, or other actors that exist in the real social world that is being modeled, while the interactions between the agents can likewise correspond to the interactions between the real world actors.

Agents are generally programmed in an object-oriented programming language and using some special-purpose simulation library or modeling environment, and are constructed using collections of condition-action rules to be able to perceive and react to their situation, to pursue the goals they are given, and to interact with other agents, for example by exchanging messages. Many hundreds of multi-agent social simulation models have now been designed and built to examine a very wide range of social and economic phenomena. For instance, urban water management ([4], [8]), institutional design [9], [13] or stock markets [10].

Like deduction, agent-based social simulation starts with a set of explicit assumptions. But unlike deduction, it does not prove theorems. Instead, a simulation generates data that can be inductively analyzed. Unlike typical induction, however, the simulated data comes from a rigorously specified set of rules rather than direct measurement of the real world. While induction can be used to find patterns in data, and deduction can be used to find consequences of assumptions, simulation modeling can be used as an aid intuition.

ABM allows modelers to complement non-formal models, usually expressed in natural language, with computer models, which are more formal. This combination avoids, at least partially, including brave assumptions to make it analytically tractable. One of the main advantages of ABM, and in our opinion the one that distinguishes it from other modeling paradigms is that it facilitates a direct correspondence between the entities in the target system and the parts of the computational model that represent them (i.e. the agents).

The process of abstraction to transform the real target system into a simulation model involves different subtasks and roles, which need diverse backgrounds and competences in the design, implementation and use of an archetypical agent based simulation.

The modeler's job is to produce formal requirements for the models starting from the thematician’s ideas. These requirements allow the computer scientist to formulate a feasible model that can run in a computer. However, not all the formal specifications can be directly implemented in a computer. The computer scientist role finds a suitable approximation to the modeler's formal model that can be executed in a computational system with the available technology. Finally, the programmer’s role is to implement the computer scientist's model to a target simulation platform.

In practical terms, modeling in social sciences faces two problems:

1. In individual developments, it is difficult that one person has all the expertise required; in teams, there are communication problems. The first problem appears when the same person plays all the roles of the process. As a consequence, many home-grown computational experimental tools are (from a software engineering perspective) poorly designed”. Besides, scientists may find difficult understanding the detailed behavior of the underlying software, since doing it would imply a full understanding of its implementation.

On the other hand, many problems require a multidisciplinary perspective, involving members with specialized roles.

2. The second problem arises because effective communication between experts of fundamentally different domains (e.g. sociology and computer science) is not trivial. In most cases, it is difficult to grasp how the social features have been mapped to program constructions. Thus, there are difficulties to assure that the program really implements its conceptual model.

To address these problems, our research promotes the creation of a set of high-level tools, methods and languages, to assist the transfer of models between different roles in the modeling process. These tools should work with modeling languages that include, ideally, concepts close to the thematician’s background, but at the same time representing ideas from a software engineering point of view. We must take into account that any mismatch between the specifications and the actual model passed to the next stage, will end up producing an error.

3 Complex Social Systems

Some recent developments in Social Sciences [7] claim for the usefulness of both experimentation (such the Experimental Economics research stream) and computer simulation as *generative* methods to analyze the complexity of social systems.
A complex social system consists of a collection of individuals that directly interact among them or through their social and technological environment. These individuals own a set of attributes that autonomously evolve, are motivated by their own beliefs and personal goals, and act under the specific circumstances of their social environment. This environment also contributes to shape their beliefs (values and knowledge about the world), can evolve in time, and has a complex structure so it is not easy to predict its net effects because of the different influence of each kind of interaction context. It has also to be taken into account that socio-economic phenomena are contingent, so they are unpredictable and changing.
The economy is complex because we observe just real aggregated data that comes from simple agents, myriad interactions. So that from a macro observation post the mathematical approach may be very difficult or computationally intractable. But if we adopt a micro definition of even heterogeneous but human agents, the resulting MAS model may be generated in a reasonable and realistic way.
Universal laws that have been used in explanations for a *general* or *average individual* often prove to be inappropriate when modeling complex social systems. The problem is that no one behaves like this *average* person. All these facts contribute to make social systems highly dynamic and complex. For this reason, abstracting them to functional mathematical models (by using, for instance, structural equation modeling, multivariate statistical analysis or statistical processing of temporal series) should be complemented by other techniques that consider how global and emergent behavior can be derived from the real subjects' behaviors, which are fundamental in any social system.
Yet another problem linked with the formalization of complex social systems is the selection of the *basic* elements to include; that is because formalization is a kind of reduction or simplification. Recent cognitive and decisional models of interacting social agents claim for the inclusion of a wider range of attributes than those used by early computational economics in the pioneer socio-economic simulations. So there is a strong challenge in the social science domain dealing with the *theoretical justification* of the minimum set of attributes that can be considered constitutive of a *social agent*.
The representative agent is not a realistic assumption to start with. We have to deal with bounded rational agents, with finite processing capacity and without explicit utility functions. They adapt and settle for satisfaction under rules of thumb. They have emotions. And they are rather heterogeneous. Even if the resulting model with a representative full rational agent has high predictive capacity, it is still important to replicate the observed patterns from models with heterogeneous and bounded rational agents.

4 Multi Agent Systems

On the other hand, from a software engineering point of view, a multi-agent system (MAS) consists of a set of autonomous software entities (the agents) that interact among them and with their environment. Autonomy means that agents are active entities that can take their own decisions. In this sense, the agent paradigm assimilates quite well the individual in a social system. In fact, there are numerous works in agent theory on organizational issues of MAS. Besides, theories from the field of Psychology have been incorporated

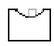
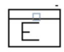
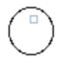
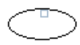
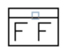
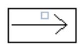
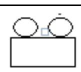
to design agent behavior, being the most extended the *Believes-Desires-Intentions* (BDI) model, on the work of Bratman [3].
With this interlinked perspective, agent-based simulation tools have been developed in the last years to explore the complexity of dynamics of complex social systems. An agent-based simulation executes several agents, which can be of different types, in an observable environment where agents' behavior can be monitored. Observations on agents can assist in the analysis of the evolution of their mental state (that is, individual values and *reasons to act*), the collective behavior and the general trends of system evolution. This provides a platform for empirical studies of social systems evolution. As simulation is performed in a controlled environment, on one or several processors, this kind of tools allows the implementation of experiments and studies that would not be feasible otherwise [5].
There are, however, some limitations when trying to simulate real social systems. The main issue is that individuals, with regard to a software agent, are by nature complex systems, whose behavior is unpredictable and less determined than for a software agent, whose behavior and perception capabilities can be designed with relative simplicity. Moreover, it is not possible in practice to consider the simulation of countless nuances that can be found in a real social system with respect to agent interaction, characterization of the environment, etc. For this reason, it is impractical to intend the simulation of a social system in all dimensions. On the other hand, we should and can limit to simulate concrete social processes in a systemic and interactive context. Therefore, the simulation of social systems should be considered in terms of focus on a concrete process under research attention.
In spite of these limitations, the agent paradigm offers many advantages to express the nature and peculiarities of social phenomena, and to overcome limitations of statistical modeling. However, social scientists that want to use this new methodology must confront a difficulty of practical order that should not be minimized. The use of existing agent based simulation tools is not simple because models have to be specified as programs, usually with an object-oriented programming language. This makes the definition of models a complex task for sociologists and other social scientists and professionals, as usually they have not developed the skills for computer programming.
That is the main reason why some tools start to offer some graphical modeling capabilities. For instance, SeSam (www.simsesam.de) allows the graphical specification of state machines and provides a library of basic behaviors. In addition, Repast Py (repast.sourceforge.net/repastpy) facilitates the visual construction of simple simulations out of some component pieces, although at the end the user needs to write Python scripts. The problem with these solutions is that they also require some programming skills and the type of systems that can be modeled are quite simple (they are mainly rapid prototyping tools).
Agent-oriented software engineering, however, offers powerful modeling languages, at a more abstract level. Concepts in these languages are closer to those that a social scientist would use, and this makes them more appropriate to solve this usability issue [15].

5 A model driven approach for agent based simulation of complex social systems

With this working hypothesis, we are developing an agent-based modeling and simulation framework by extending a concrete agent-oriented methodology, INGENIAS [12]. This framework will allow the specification of social systems with a graphical modeling language, the simulation of the models of these systems by exploiting the capabilities of existing agent-based simulation tools/platforms, and the identification and analysis of social patterns (at a macroscopic, or aggregate, level) in terms of the atomic elements of the social system specification (at a microscopic, or individual/interaction, level). The advantages

go further than usability. As it has been discussed in [14], this solution facilitates the replication of an experiment on different simulation engines, in order to contrast results. The availability of a graphical view of the system facilitates its understanding too and improves the identification of patterns in the system [11].
There are two main reasons for the choice of INGENIAS as starting point for this work. First, its modeling language supports well the specification of organization structure and dynamics, as well as agent intentional behavior, characteristics that are present in social systems [6]. This language is supported by the INGENIAS Development Kit (IDK) with a graphical editor, which can be extended to introduce new modeling concepts. Second, INGENIAS promotes a model-driven engineering approach that facilitates the independence of the modeling language with respect to the implementation platform. This is especially important here in order to abstract away programming details and concentrate on modeling and analysis of social patterns. With this purpose the IDK supports the definition of transformations between models and code for a range of implementation platforms.
For ABM is particularly relevant the availability of graphical editors to define these metamodels and to generate from them customized editors for DSL (Domain-Specific-Languages). This facilitates providing researchers with specific modeling tools for their domains, which improve their productivity and reduce the probability of mistakes.

Table 1. Main concepts of the INGENIAS modeling language

Concept	Meaning	Icons
Agent	An active concept with explicit goals that is able to initiate some actions involving other elements of the simulation.	
Role	A role groups related goals and tasks. An agent playing a role acquires the goals and tasks of such role.	
Environment application	An element of the environment. Agents act on the environment using its actions and receive information through its events.	
Goal	An objective of an agent. Agents try to satisfy their goals executing tasks. The satisfaction or failure of a goal depends on the presence or absence of some elements (i.e. frame facts and events) in the society or the environment.	
Task	A capability of an agent. In order to execute a task, certain elements (i.e. frame facts and events) must be available. The execution produces/consumes some elements as result.	
Frame Fact	An element produced by a task, and therefore by the agents.	
Event	An element produced by an environment application.	
Interaction	Any kind of social activity involving several agents.	
Group	A set of agents that share some common goals and the applications they have access to.	
Society	A set of agents, applications and groups, along with some general rules that govern the agent and group behavior.	

6 Conclusions

Although in the last years there is extensive experience on the development of agent-based simulation toolkits, these tools are still rather difficult to be used by typical social scientists and economists, as they require skills in computer programming languages. In order to overcome this issue, we propose to wrap these tools with modeling tools. These tools rely on an agent based modeling language, which provides facilities to specify complex social systems, their deployment on simulation platforms, and analysis of simulation results. This language is specified as a set of meta-models, which are used as the foundation for building the tools. This has the advantage that tools can be easily customized for particular problem domains. The purpose therefore is not to provide a single general tool, but customized toolkits for each sociological problem domain. This is achieved by adaptation of the meta-models to the conceptual needs of each problem domain (this means, to facilitate the representation with the modeling language of the concepts that are required on the problem domain). In this way, language and tools usability is increased as they are adapted for specific purposes. Tools comprise a graphical model editor, document and code generation, analysis and validation. Code generation is based on generative programming techniques, which have been already applied in INGENIAS Development Kit (IDK). In fact, the IDK provides an agent-oriented modeling language that can be the basis for complex social system modeling languages, and a framework for building code generation and analysis tools.

Acknowledgments

This work is done in the context of the following projects: (1) “Agent-based Modeling and Simulation of Complex Social Systems (SiCoSSys)”, supported by Spanish Council for Science and Innovation, with grants TIN2008-06464-C03-01 and TIN2008-06464-C03-02; (2) “ABACO VA006A09”, (3) the Programa de Creación y Consolidación de Grupos de Investigación UCM-BSCH GR58-08, and (4) GR251/09 supported by the “Junta de Castilla y Leon”.

References

[1] R. Axelrod, “Advancing the Art of Simulation in the Social Sciences,” *in Rosaria Conte, Rainer Hegselmann and Pietro Terna (eds.), Simulating Social Phenomena*, Berlin, Springer, 1997, pp. 21-40.

[2] R. J. Barkley, “On the complexities of complex economic dynamics,” *Journal of Economic Perspectives*, Vol. 13, No. 4, 1999, pp. 169–192.

[3] M. E. Bratman, *Intentions, Plans and Practical Reason*, Harvard University Press, 1987.

[4] J. M. Galán, A. López-Paredes and R. del Olmo, “An agent-based model for domestic water management in Valladolid metropolitan area,” *Water Resources Research*, No. 45, W05401, 2009.

[5] N. Gilbert and K. Troitzsch, *Simulación para las Ciencias Sociales: Una guía práctica para explorar cuestiones sociales mediante el uso de simulaciones informáticas (2ª edición)*, Madrid, McGraw-Hill, 2006.

[6] J. Gómez-Sanz, J. Pavón and F. Garijo, “Meta-models for Building Multi-Agent Systems,” *The 2002 ACM Symposium on Applied Computing (SAC 2002)*, ACM Press, 2002, pp. 37-41.

[7] P. Hedström, *Dissecting the Social: On the Principles of Analytical Sociology*, Cambridge University Press, 2005.

[8] A. López-Paredes, D. Saurí and J. M. Galán, “Urban Water Management with Artificial Societies of Agents. The FIRMABAR Simulator,” *Simulation-Transactions of the Society of Modelling and Simulation International*, Vol. 81, No. 3, 2005, pp. 189-199.

[9] A. López-Paredes, C. Hernández, and J. Pajares, “Towards a New Experimental Socio-economics. Complex Behaviour in Bargaining,” *Journal of Socioeconomics*, Vol. 31, Elsevier, 2002, pp. 423 – 429.

[10] J. A. Pascual, J. Pajares and A. López Paredes, “Explaining the Statistical Features of the Spanish Stock Market from the Bottom-Up,” *In Lecture Notes in Economics and Mathematical Systems. Advances in Artificial Economics*, No. 584, 2006, pp. 283-294.

[11] J. Pavón, M. Arroyo, S. Hassan and C. Sansores, “Agent Based Modelling and Simulation for the Analysis of Social Patterns,” 2008, doi:10.1016/j.patrec.2007.06.021.

[12] J. Pavón, J. J. Gómez-Sanz and R. Fuentes, “The INGENIAS Methodology and Tools. In: Agent-Oriented Methodologies,” *Idea Group*

Publishing, 2005, pp. 236-276.

[13] M. Posada and A. López-Paredes, "How to Choose the Bidding Strategy in Continuous Double Auctions: Imitation Versus Take-The-Best Heuristics," *Journal of Artificial Societies and Social Simulation*, Vol. 11, No. 1, 6 SIMSOC Consortium, 2008.

[14] C. Sansores and J. Pavón, "Simulación social basada en agentes," *Revista Iberoamericana de Inteligencia Artificial*, No. 25, 2004, pp. 71-78.

[15] F. Squazzoni and R. Boero, *Towards an Agent-Based Computational Sociology. Good Reasons to Strengthen a Cross-Fertilization Between Complexity and Sociology*, in (ed) Stoneham P. M., Advances in Sociology Research II, Nova Science Publishers Inc., New York, USA, 2005, pp. 103-133.

Adolfo LÓPEZ-PAREDES is Associate Professor at the University of Valladolid (Spain). He graduated from the University of Oviedo in Industrial Engineering (1994) and obtained his Ph.D. in 2000 on applications of agent-based simulation to economic analysis, in the University of the Basque Country. His research interests within the INSISOC Group ("Engineering Social Systems Group") include project management and computer simulation of social and economic behaviour and industrial policy: auctions, financial markets, natural resources management, supply chain management.

Juan PAVÓN holds a PhD degree in Computer Science from Universidad Politécnica Madrid (1988) and since 1997 is Associate Professor at Universidad Complutense of Madrid. From 1987 to 1997 he was working in R&D departments of Alcatel in Spain, France and Belgium, and in Bellcore (USA), especially in the development of component-based architectures for distributed systems, and their application to multimedia services on broadband networks and mobile systems. Currently he is the leader of the Grasia research group at Universidad Complutense of Madrid, where he has been involved in several research projects on the application of multi-agent systems technology, in particular, on software engineering, distributed control, web services personalization, knowledge management, and social simulation.