Intentional Adequacy of Computer Programs as the Experimental Reference of Agent-Based Social Simulation

Nuno David
ISCTE/DCTI, Department of Information Science and Technology
Av. das Forças Armadas
1649-026 Lisboa, Portugal
Nuno.David@iscte.pt

Jaime Simão Sichman[1]
Intelligent Techniques Laboratory
University of São Paulo
Av. Prof. Luciano Gualberto, tv.3, 158,
05508-900 Sao Paulo SP Brazil
Jaime.Sichman@poli.usp.br

Helder Coelho
Department of Informatics
University of Lisbon
Bloco C6, Piso 3 (06), 1749-016
Lisboa, Portugal
hcoelho@di.fc.ul.pt

ABSTRACT

The classical theory of computation is not an adequate model of reality for agent-based simulation in the social sciences. The paradigm of intentional computation seems to be the only one possible to reflect the multiparadigmatic character of social science in terms of agent-based computational social science. This is a paradigm that enlarges the concept of valid computation, which must be dependent on the particular theoretical-methodological context of the social scientist.

Categories and Subject Descriptors

I.6.0 [**Simulation and Modeling**]: General; I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence -- *multiagent systems, intelligent agents.*

General Terms

Design, Experimentation, Human Factors, Theory, Verification.

Keywords

Social Simulation, Programming, Intentional Verification.

1. SIMULATION AND KNOWLEDGE

The role of computer simulation has acquired a renewed importance in the social sciences. From an interdisciplinary perspective, the simulation of social theories and phenomena finds its origin in the intersection of the social and computer sciences. In particular, after the consolidation of the multiagent paradigm in Artificial Intelligence, the discipline of Agent-Based Social Simulation has promoted the reunion of two broad scientific logics with distinct methodological grounds: on the one hand, the formal and empirical logic of computer science and, on the other hand, the descriptive and interpretative logic of the social sciences. Whereas from an interdisciplinary viewpoint the discipline stresses the meeting of two distinct scientific logics, there are undoubtedly good reasons to maintain methodology in the research agenda.

For some, the use of formal models resulting from the computational nature of simulation has been considered not only an addition to the established methods but the basis for the emergence of proper social sciences. Even so, the difficulties in constructing and analyzing simulations, even the most simplified, have been underlined in the literature, which raises some interesting questions around the kind of scientific knowledge that simulation is providing.

There are at least two aspects that have not received sufficient analysis. Firstly, the *experimental reference* of simulation remains ambiguous, insofar as the logic of its method turns computer programs into something more than a tool in the social sciences, defining them as the experimental object itself – it is programs, and not the social phenomena they presumably represent, that are executed and tested. Secondly, the *formal tradition* of the classic theory of computation, derived from the Church-Turing thesis, creates a semantic gap between the formal interpretation of program execution behaviors and the stakeholders' interpretations acquired via informal observation of simulations.

Indeed, a computational theory of simulation will hardly be founded upon the classic theory of computation. If we accept the Church-Turing thesis, then it is legitimate to say that all computation that terminates can be simulated by a first-order language. But insofar as we find the expressiveness of first-order languages insufficient to describe target social phenomena and theory, the description of computational processes in terms of the interpretative character of the social sciences finds serious obstacles.

While some of these difficulties have deserved discussion in the field, they suffer from the absence of foundational visions as to the role played by social simulation in scientific knowledge. How are we to reconcile the methodologically diverse and multiparadigmatic social sciences with a computer science that has been able to attain a larger consensus in regard to the conception of scientific truth or validity? Insofar as the behavior of computer programs is analyzed in terms of the interpretative character of the social sciences, we should attempt to evaluate whether simulation implies an additional perspective in the way we understand the kind of computation, and knowledge, that simulation is providing.

After the SimCog survey [2], it became clear to us that the understanding of a particular simulation involves a dialectical approach between four factors: *intention*, *experimentation*, *interpretation* and *diversity*. Those can be described as follows: (i) the social scientist's *intention* in building a particular simulation technology (i.e. the intention in implementing a

AAMAS'05, July 25-29, 2005, Utrecht, Netherlands.

particular program); (ii) the *experimentation* processes (i.e. how to test the adequacy, or *intended* meanings, of the behavior of the program); (iii) the *interpretation* of results in terms of the particular theoretical-methodological context of the social scientist (i.e. the actual *interpretation* of the behavior of the program); and (iv) the *diversity* of theoretical-methodological contexts in the social sciences, as well as the specific context of the simulation stakeholders (i.e. the subjective meaning of the behavior of the program).

The link between these factors motivated us to investigate the experimental nature of social simulation. Our aim is to construct a comprehensive methodological perspective that should be able to conciliate the formal and empirical logic of program verification in computer science with the diverse and interpretative logic of the social sciences.

In conclusion, our analysis revealed that simulation programs seem to possess an intentional capability that classical programs do not. The intentional capability of programs does not resemble Fetzer's concept of causal capability of programs [3]. This distinction is associated with two types of experimental verification of programs, called empirical and intentional verification. This means that the logic of social simulation reflects a distinction in the kind of experimental knowledge one can have about programs. It also means that the classical theory of computation is not an adequate model of reality for computational social science.

2. THE STRUCTURE OF OUR ARGUMENT

Given the limited space in this abstract, our goal is to outline very briefly the structure of our argument. The whole argument can be found in [1]. It is comprised of three parts.

First Part: FDE refutation

The first part consists of characterizing the unsatisfactory role of the classical theory of computation in the complexity sciences, particularly in social simulation. Traditional scientific methodologies often characterize the concept of program execution as a process of formal inference. To some extent, this recalcitrant tradition results from conflating the terms "program computation" and "program execution" into one single meaning, conveying the same ontological status to two fundamentally distinct processes.

Considerations of brevity led us to call this tradition the FDE argument: *Formal Deduction through Execution*. Nevertheless, our goal is quite the opposite, namely to show that simulation should not be legitimized under the presumption of being a result of a calculus of formal inference. The FDE refutation suggests yet another objection to current philosophical thinking in the literature, namely the characterization of simulation as a new basic epistemic conception of scientific methodology, such as deduction or induction. Whereas social simulation seems to represent a new kind of experimental science, the multiparadigmatic character of the social sciences requires that simulation be understood beyond traditional characterizations of formal and empirical sciences.

Second Part: The role of programming languages in simulation

The second part of our argument analyses the role of programming languages as embedded models in the simulations. The goal is to describe methodological aspects of simulation that may or may not be characterized within the scope of an empirical science. We first recall that computer programs have a semantic significance related to their causal capability that scientific theories do not seem to possess. However, whereas the formal and causal roles of programming languages may be well characterized in classical computation [3], the informal role of social simulation languages has a distinctive function.

The informal character of social simulation suggests that its methods highlight the presence of more outstanding intentional aspects in programming and interaction with computers than might be expected. Despite the fact that the results of simulation are outcomes of experimental processes, they do not represent "material conditions of necessity" between facts about program behaviors. Instead, the analysis of "conditions of intentionality," led us to the third and final part of our argument.

Third Part: The double interpretative character of simulation

One way to understand the role of experimentation in social simulation is to realize that theories in simulation are *doubly contingent*, i.e. interpreted according to two distinct references: the program behavior itself and the targeted social theory or phenomenon. Note, however, that the former cannot be interpreted as a model of the latter, at least within the context of the classical theory of computation. Insofar as social theories and phenomena are hardly described by first-order languages, it is possible to show that there are two complementary scientific logics at stake in social simulation. One is based on the formal and empirical logic of program verification in which *necessary conditions* about program behaviors are specified and verified empirically. Another is based on the experimental logic of program verification in which *intentionality conditions* about program behaviors are specified and verified experimentally, albeit not empirically, according to limited consensus and limited communities of observers. This distinction is associated with two kinds of program verification, called empirical and intentional verification.

In conclusion, the experimental role of simulation suggests a further context of scientific knowledge, beyond the traditional division between formal and empirical knowledge: *formal*, *empirical* and *intentional* knowledge. The paradigm of intentional computation seems to be the only one possible to reflect the multiparadigmatic character of social science into agent-based computational social science. This is a kind of computation that enlarges the concept of valid computation, which should be dependent on the particular theoretical-methodological context of the social scientist.

3. REFERENCES

[1] David, N. (2005). *Veriricação Empírica e Intencional de Programas em Simulação Social Baseada em Agentes*. PhD Thesis. University of Lisbon (in Portuguese).

[2] David, N.; Marietto, M.; Sichman, JS. and Coelho, H (2004). *The Structure and Logic of Interdisciplinary Research in Agent-Based Social Simulation*. Journal of Artificial Societies and Social Simulation (JASSS), v.7, n.3.

[3] Fetzer, J. (1988). *Program Verification: The Very Idea*. Communications of the ACM, v.31, pp.1048-1063.

[1] Partially supported by CNPq, Brazil, grant 304605/2004-2.