Evolutionary Programming in SADDE

Carles Sierra
IIIA - CSIC, Campus UAB,
Bellaterra, Barcelona, Spain
sierra@iiia.csic.es

Jordi Sabater
IIIA - CSIC, Campus UAB,
Bellaterra, Barcelona, Spain
jsabater@iiia.csic.es

Jaume Agustí
IIIA - CSIC, Campus UAB,
Bellaterra, Barcelona, Spain
agusti@iiia.csic.es

Categories and Subject Descriptors

I.2 [**Computing Methodologies**]: Artificial Intelligence; D.2 [**Software**]: Software Engineering; I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*intelligent agents, multiagent systems*

General Terms

Experimentation

1. INTRODUCTION

The general goal of the research reported in this paper is to better understand the dynamics of large Multi-Agent Systems (or MAS, for short) with globally distributed and interconnected collections of human, software and hardware systems; each one of which with potentially thousands of components.

This paper explores two ideas. First, a particular approach to the principled design of MAS using Equation-Based Models (EBM) as a high level specification method, where equations model the aggregated behaviour of the agent populations abstracting from the interaction details of individual agents (Section 2). Second, the use of evolutionary computation techniques to find out what agent structures produce the global emergent behaviour specified in the EBM (Section 3). These ideas are framed within a design methodology called SADDE (Social Agents Design Driven by Equations) [3].

2. THE SADDE METHODOLOGY

We take the stance that in order to build a model for a society containing thousands or millions of agents, the general view provided by an EBM provides succint descriptions of population-level behaviours which we then attempt to replicate using models consisting of a society of individual interacting agents, that is, the ABM (Figure 1).

An important characteristic of MASs design from a software engineering perspective is the decoupling of the interaction process between agents from the deliberative/reactive

AAMAS'02, July 15-19, 2002, Bologna, Italy.

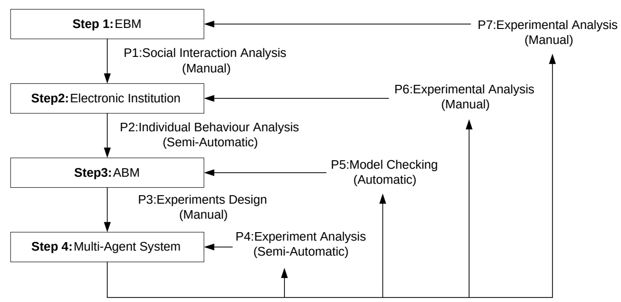

Figure 1: SADDE Methodology

activity within each agent. The notion of *electronic institution* [1], plays this role in our methodology by establishing a framework that constraints and enforces the acceptable behaviour of agents.

The different phases within SADDE are:

[Step 1] **EBM – Equation-Based Model**. In this first step, a set of state variables and equations relating them must be identified. These equations have to model the desired global behaviour of the agent society and will not contain references to individuals of that society. Typically these variables will refer to values in the environment and to averages of predictions for observable variables of the agents. We model yet-to-exist artificial systems. The EBM is the starting point of the construction of a system that later on will be observed. Thus, a comparison between the EBM predicted behaviour and the actual ABM behaviour will be obtained.

[Step 2] **EIM – Electronic Institution Model**. In this step the interactions among agents are the focus. It is a first "zoom in" of the methodology from the global view towards the individual models. This step is not a refinement of the EBM but the design of a set of social interaction norms that are consistent with the relations established at Step 1.

[Step 3] **ABM – Agent-Based Model**. Here, we focus in the individual. We have to decide what decision models to use. This is the second "zoom in" of the methodology. New elements of the requirement analysis (new variables) will be taken into account here. For instance, some rationality principles associated to agents (*e.g.* producers do not sell below production costs), or negotiation models to be used (*e.g.* as those proposed in [2]) have to be selected.

[Step 4] **Multi-Agent System**. Finally, the last step of our methodology consists on the design of experiments for the interaction of very large numbers of agents designed in the previous step. For each type of agent the number of individuals and the concrete setting for the parameters

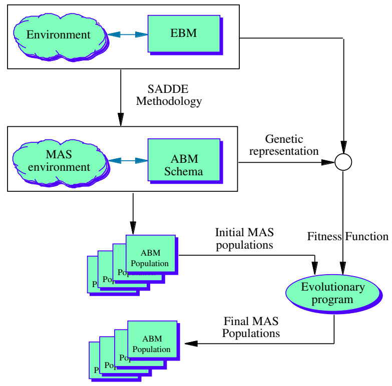

Figure 2: Evolutionary computing within SADDE.

will be the matter of decision here. The results of these experiments will determine whether the requirements of the artificial society so constructed have been consistently interpreted throughout the methodology and thus whether the expected results according to the EBM are confirmed or not.

Once the experiments designed at Step 4 are run and analysed, several redesigns are possible as shown schematically in figure 1. In next section we focus on the use of evolutionary computation to explore the space of possible MAS configurations.

3. EVOLUTION AND SADDE

Once an ABM is generated according to the SADDE methodology, what we have is a precisely defined way of interaction between agents, as restricted by the electronic Institution, and schemes of individual behaviour (determined by a concrete decision making system) of the agents playing the different roles. But there are still two important decisions to be made in order to have a running MAS: what values to assign to the parameters of the decision making apparatus of the agents, and what proportions of significantly different individual behaviours to use in order to conform to the MAS.

Evolutionary computing is the technique we are currentky using to explore the space of possible configurations of MAS populations. An individual in the Genetic Algorithm corresponds, in our case, to the genetic material of a complete MAS population. Crossover between populations will mean that subsequent generations will explore the space of agent combinations and that mutation will, basically, generate new agents by mutating the parameters of the decision making apparatus of a particular agent. From the study of such populations we expect to obtain insights about the structure of the agents and their social interelationships in relation to the global behaviour. This study permits the definition of a series of design rules that help reducing the currently existing gap between specification and implementation of MAS.

Figure 2 illustrates graphically the intended role of evolutionary computation. We want to use as the initial population of the evolutionary computation algorithm a set of MASs that fit with the schema obtained through the SADDE methodology, and then use evolutionary computation to obtain a set of MASs that fit optimally with the EBM. It is thus natural that the fitness function is provided by the EBM in terms of the concrete genetic coding used. In our approach, one chromosome is the specification of a full MAS and the parameters that specify a single agent are codified in a gen of that chromosome.

3.1 Fitness functions

One of the key design issues in the proposed methodology is how to obtain a fitness function from the global behaviour, as expressed in the EBM, and from the behaviour of the agents, as specified in the ABM. The right choice is essential to improve from the initially designed ABM populations into better ABM populations that fit the overall objective of guaranteeing certain desired properties of the societies satisfying the EBM.

Thus, in order to determine the fitness function we put in relation these global properties with individual variables so that by selecting MASs that maximize some functions over those variables we approach the desired global behaviour. In general, if we have a set of properties we want the MAS to satisfy along time and we model each property to be satisfied as a function over time and a vector of state variables in the EBM, $f_i(t, \mathbf{X}_i)$, and we model the observed behaviour of the aggregated individual variables, $\mathbf{Y}_i$, corresponding to $\mathbf{X}_i$, as $h_i(t, \mathbf{Y}_i)$, we can define a fitness function as a weighted ($omega_i$) mean over a comparison function (v.g. quadratic means error) between the two along time:

$$f(EBM, ABM) = \sum_{i=1}^{n} \omega_i \cdot \sum_{0 \leq t \leq T} g(f_i(t, \mathbf{X}_i), h_i(t, \mathbf{Y}_i))$$

4. DISCUSSION AND RESULTS

EBM and ABM are two well known styles of computer based modelling. We have integrated both approaches into a methodology for MAS design and implementation. More specifically we have used EBM to identify desired global properties of the MAS.

The application of GA techniques to a collection of MAS in the context of a supply chain example brought us two main preliminary results. First, the chosen agent model allowed the convergence of the evolutionary process towards the production of a stable collection of MASs showing the EBM specified properties —referring evolution of cash and stock— to an acceptable degree. Second, from the analysis of the distribution of the values of the parameters of the negotiation model used by the agent in each MAS we have established several design rules which relate them with the global properties specified by the EBM.

5. ACKNOWLEDGEMENTS

Research supported by the CICYT project eINSTITUTOR (TIC2000-1414), and by the European research project SLIE (IST-1999-10948).

6. ADDITIONAL AUTHORS

Additional authors: Pere Garcia (IIIA - CSIC, Campus UAB, Bellaterra, email: `pere@iiia.csic.es`)

7. REFERENCES

[1] P. Noriega and C. Sierra, *Auctions and Multi-agent Systems*, in Matthias Klush (ed.): 'Intelligent Information Agents', pp. 153–175, Springer, (1999).

[2] C. Sierra, P. Faratin, and N.R. Jennings, 'A service-oriented negotiation model between autonomous agents', in *MAAMAW'97*, number 1237 in LNAI, pp. 17–35, Ronneby, Sweden, (1997).

[3] C. Sierra, J. Sabater, J. Agustí, and P. Garcia, 'Evolutionary Computation in MAS Design', in *Proc. ECAI*, in press, (2002).