Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 1231–1238
ISBN 978-83-60810-51-4

Hybrid Multi-agent system simulations: cognitive and social agents

Alberto Caballero
Universidad Católica San Antonio
Guadalupe, Murcia 30107
Email: acaballero@ucam.edu

Juan Botía
Universidad de Murcia
Espinardo, Murcia 30100
Email: juanbot@um.es

***Abstract*—Simulating social and cognitive agent abilities is a very important aspect of agent-based computing. Multi-Agent Based Social Simulations (MABSS) could benefit from incorporating cognitive behaviours. A hybrid simulating approach, considering social and cognitive abilities, provides a more realistic basis for modelling agents and their social interactions. But, how social and cognitive behaviours could be supported simultaneously in MABSS? Is it always advantageous using cognitive capabilities into social simulations? This paper offers a set of general considerations about how cognitive capabilities could be integrated into social multi-agent simulations. It points out the most relevant cognitive requirements of social simulations of a great amount of real scenarios where some agents could carry out cognitive processing while others (a great majority) behave in reactive way. The suitability of several alternatives for integrating social and cognitive capabilities of agents are discussed. The paper also offers several efficiency related arguments and recommendations for using one of the three considered approaches.**

I. Introduction

MULTI-AGENT Based Social Simulations (MABSS) is a paradigm devoted to use agents as the modeling metaphor to simulate autonomous entities in a social world composed by a number of independent and interacting entities [1]. Such models try to reproduce real environments and situations of interest within such environments. Generally, in these environments, some agents could carry out cognitive processing while others (a great majority) behave in a reactive way [2], [3], [4], [5]. Both types of agents do not have the same requirements. For instance, the cognitive agents do not perform cognitive processing in every moments. In front of cognitive agents, each reactive one could modify its own state in every step of the simulation.

In addition, all cognitive agents do not have the same requirements in all type of scenarios. For instance, sometimes several reasoning or explaining process are needed (e.g living labs [5] and large complex smart grids through broker agents [6]). In other scenarios, like Ubik [3] or Cardinea [4], several planning capabilities are exhibited by agents in order to plan the escaping out the building or to organize the schedule of an employee, respectively.

In Cardinea [4], there are several agents representing supervisors and assistants in a hospital. Only supervisors could carry out cognitive processing while the assistants (a great majority of them) behave in a reactive way. Sometimes, when a given assistant reaches the limit of contamination, his supervisor needs to carry out some deliberative processing in order to decide what the assistant must to do. In this case, an agent architecture could be required to perceive the knowledge related to the assistant and to reason about that. Communicative requirements of this scenario are very simple: supervisors only need to inform to other supervisors, in some (not all) steps, the actions that assistants must to do.

Other scenarios, such as the smart grids through broker agents [6], could require support for more complex dialogs. In these scenarios, agents interact in order to discover, to evaluate and to learn market strategies aiming to earn profits while maintaining the markets balance of supply and demand.

On the other hand, several platforms for MABSS are available for exploring the behaviour of agents. All of these tools incorporate a set of functional capabilities to simulate multiple interactions between several agents, to collect data about the performance of the agents, a set of interfaces for developing, setting up the parameters, monitoring and controlling the simulations, etc. However, the way in which the agent's requirements are satisfied may be different.

Some of these platforms, the social ones, are oriented to simulate simple reactive agents, with low communicative requirements (e.g. Mason [7], NetLogo [1], Ascape [8]), while others, the cognitive ones, are focused to support complex cognitive agents's behaviours (e.g. Soar [9], ACT-R [10], 4CAPS [11], CLARION [12]). Others offer hybrids solutions to support social and cognitive agents abilities in the same tool [13], [14], [15].

All platforms are not equally useful for each scenario. Generally, the social platforms are fast, simple and easy to learn, and they not require new programming abilities. However, incorporating cognitive capabilities in the simulated agents could be not easy. Moreover, the cognitive platforms provide cognitive capabilities to social agents in a suitable way but they are not able to simulate swarm systems. Each one separately could not be able to perform cognitive processing in wide simulations, as Cardinea requires. Hence, a hybrid approach could be very useful. The quantitative analysis of the performance of a hybrid approach (while the suitability of this approach to medium-demand scenario is evaluated) could help to offer qualitative valuations about the suitability of the social, cognitive and hybrid tools for other (low- or high-demand) scenarios.

Hence, it would be interesting to offer recommendations for use, taking into account the suitability of each MABSS platform.

The main contributions of this paper are related with (1) the identification of several cognitive requirements in social simulations where some agents show cognitive behaviours and a great majority of them behaves in a reactive way, (2) the quantitative evaluation of the suitability of a MABSS hybrid tool to simulate agents with social and cognitive abilities, and, (3) the proposal of several recommendations for use of a particular MABSS for a specific scenario whose cognitive and/or social requirements have been previously identified. Possibly, the last one is the most valuable and generally applicable outcome of this paper for the research community in the MABSS domain.

The rest of the paper is structured as follow: section II identifies the most relevant requirements of social and cognitive agents in MABSS. Several simulating platforms of different types (social, cognitive and hybrid) are referenced in section III. An empirical evaluation about the suitability of two hybrid approaches is offered in section IV. Based on that, several recommendations for using each MABSS platform according to the social and cognitive requirements are presented in section V. Finally, section VI outlines conclusions and some ideas for future work.

II. Requirements of MABSS

Actually, there are several MABSS platforms used to simulate social and cognitive agent behaviours into a wide variety of scenarios (e.g. MASON [7], RePast [16], NetLogo [1], SeSAm [17]). Agents simulated in these scenarios could be demand very different functionalities to the MABSS platforms. However, several common requirements are highlighted. The most relevant ones are listed in the table I.

III. Platforms for MABSS

Three kinds of MABSS platforms could be identified taking into account their capabilities for representing and exploring social and cognitive agent behaviours: (1) social, (2) cognitive, and (3) hybrid ones. In the following, the most relevant platforms of each kind of platforms and their suitability, in order to satisfy the MABSS requirements identified in the section II, are also related.

A. Social platforms

Nowadays, most used MABSS platforms offer easy to use interfaces so that scientists can develop simulations even without considering cognitive abilities of the agents [18]. This kind of tools is very useful when simulated agents need to satisfy the requirements 1, 2, 7 and 9 (listed in table I). Some of them also report additional advantages for agent simulation. For instance, in MASON the simulation can be replicated using the same random seed. Several comparative analyses between MABSS technologies are given [19], [20]. In all of them, MASON stands out from a set of several social MABSS technologies such as SWARM [1], RePast, Ascape, NetLogo, SeSAm. This way, MASON is a suitable platform to guarantee the requirements related to the social agents abilities and interesting for experimentation.

B. Cognitive platforms

All social platforms referenced above are based on the concept of a very simple agent, totally reactive, with very low cognitive abilities. However, in a great amount of systems, attempting to reproduce intelligent human behaviours, the simulated agents show autonomy, proactivity, adaptability, reasoning, planning capabilities, and so on. This is accomplished by the so called cognitive architectures. Such systems manage decision making, memory, communication, learning, among others [9], [10], [11], [12].

They are both conceptually heavy models and intensive CPU consuming approaches. This last fact makes them unfeasible for social simulations with a high number of agents in the society. However, they are very useful when simulated agents need to satisfy the requirements 3, 4, 5 and 6 (listed in table I).

Soar, ACT-R, 4CAPS, CLARION highlight from the rest of cognitive architectures in order to provide cognitive capabilities to some agents in social simulations. Most of them use their own language to specify the knowledge of the agent or require big programming or technology familiarization efforts. For instance, several influential cognitive architectures, such as ACT-R, Soar, CLARION and 4CAPS, have computational implementations as interpreters of a special coding language.

C. Hybrid platforms

Several approaches such as Jason [21], 3APL [22] or CoJack [23] are intended for simulating a society of BDI agents. These BDI-based technologies are hybrid approaches located between the reactive social simulation platforms and the cognitive architectures. BDI architecture provides several interesting cognitive capabilities to social agents. Their conceptual complexity is lower than the offered by cognitive architectures, but it is generally still an intensive CPU consuming technology, according to the main requirements of social simulations, where a great number of social entities must be simulated.

These tools consider the integration of deliberative agents in social simulations, incorporating complex models for knowledge representation and processing. They perform complex reasoning processes, offer advanced capabilities for communication and coordination, planning, among others. However, sometimes, they could not offer the best solutions if the capabilities to replicate the simulations, the time of execution, the learning curve, the use of specific languages, among others, are taken into account.

Jason is a Java-based hybrid platform which core lies an interpreter for an extended version of AgentSpeak [24], one

[1] http://www.swarm.org

TABLE I
MOST COMMON REQUIREMENTS OF THE MABSS PLATFORMS.

For exploring social abilities:	
1) Support for swarm simulations	The great majority of simulated scenarios are characterized by the presence of a high number of agents.
2) Managing the location of the agent	Several scenarios represent bi- and tri-dimensional worlds where the location of each agent is relevant.
For exploring cognitive abilities:	
3) Using standard agent architectures	The use of a well-known agent architecture, such as the BDI, could be required in several simulations considering deliberative agents.
4) Support for reasoning, planning, explaining	If any agent architecture is used to model the agents, tools for performing cognitive processing might be required.
5) Support for communication between agents	Generally, complex tasks resulting from the coordination between several agents could be carried out.
6) Support for selecting when cognitive agents perform cognitive processing	Widely, cognitive agents do not perform cognitive tasks in all steps of the simulation.
Interesting for experimentation:	
7) Capabilities to replicate the same simulation	Frequently, replication of the simulations could be required.
8) Support for choose the type of agent requirements into simulation	All of the agents do not have the same requirements. Simulations should distinguish between cognitive and social agent's abilities.
9) Easy to use interfaces for developing, setting up the parameters, monitoring and controlling the simulations.	Suitable interfaces for coding, running and inspecting the behaviour of the simulated agents are very useful.
Performance and productivity related:	
10) Low CPU time consuming	The MABSS platforms need to be as efficient as possible, in order to simulate a great number of agents.
11) Learning curve	Generally, programmer skills are greater in general-purpose languages (such as Java) than in specific ones (such as a BDI language).
12) Programming and maintenance costs	Developing and maintenance the code of agents are more easy using a specific-purpose language than a general-purpose one.
14) Alignment with agent developing methodologies	Programming the behaviours of agents could be more easy if a formal methodologies is used to model the system.

of the best known languages based on the BDI architecture. However, Jason is not able to simulate a high number of agents that several scenarios require. Jason, similar to the cognitive architectures mentioned above, assumes that (1) all entities perform cognitive activities, and (2) these activities are carried out every time. In several scenarios, only a few number of agents needs to perform cognitive behaviours, and these behaviours are not exhibited in each timestep of the simulations.

The execution time in several hybrid platforms, like in the cognitive ones, is very high. It is basically devoted to perform cognitive processing instead of simulation controlling and data collecting. They could not be feasible for several swarm simulations, more even if a few number of cognitive ones are simulated.

This way, several hybrid proposals, combining swarming and BDI agents are made [13], [14], [15]. The approach offered in [15] is based on the addition of cognitive capabilities, supported by Jason (a BDI-based platform), to some agents in the social simulations carried out in MASON (a social platform). Hence, it takes the most advantageous characteristics of each platform. It tries to improve the CPU time consuming, since cognitive processing is only carried out when (a few number of) cognitive agents need it. Several alternatives are proposed to combine MASON and Jason processing:

1) The simulation is controlled by MASON, where Jason environment is executed like any other MASON agent.
2) The simulation is controlled by Jason, where MASON environment is executed after all Jason agents.
3) The simulation is controlled by MASON, without using Jason environment. All agents are MASON agents and only those with cognitive capabilities use functionalities of the BDI interpreter and, possibly, others supplied by a customized agent architecture.

However, from the point of view of implementation, the alternative number 1 is not feasible because neither Jason nor MASON offer the necessary programming interfaces to achieve that.

The two interesting alternatives (2 and 3) guarantee the same agent's requirements for exploring social abilities (requirements 1 and 2 listed in table I) and interesting for experimentation (requirements 7 and 9 listed in table I) since MASON is used in them. On the other hand, we suppose that all simulations demand the same requirements of the last group related to efficiency and productivity (requirements from 10 to 14 listed in table I). Hence, the performance of the two hybrid alternatives 2 and 3 and their suitability to simulate agents with social and cognitive abilities are evaluated taking into account several scenarios where agents have different cognitive requirements.

IV. SUITABILITY OF TWO HYBRID APPROACHES

This section offers evidences about the performance of the hybrid approaches 2 and 3 (presented above) for integrating MASON and Jason. Experimental evaluations commented here are focused on the study of the scalability of each approach considering a particular scenario presented in section IV-A. Three types of experiments are carried out. They are presented in the following sections:

Section IV-B: comparison between alternatives 2 and 3, in order to clarify which is the most suitable alternative to support a high number of agents where a few of them need cognitive capabilities, and non-sophisticated communication scheme.

Section IV-C: study about the performance of the alternative 2, varying the number of social and reactive agents, in order to analyze the evolution of the total simulation time and its relationship with the time performing cognitive processes and the time that message interchanging consumes.

Section IV-D: study of the performance of the alternative 3, varying the number of social and reactive agents, in order to analyze the evolution of the total simulation time and its relationship with the time performing cognitive processes.

A. Running scenario

The evaluation takes place in a particular illustrative scenario taking into account the cognitive requirements listed in table I. For that, three typical scenarios are defined. Table II shows the requirements of each typical scenario, considering the characteristics of the three real ones (Urban Traffic [2], Cardinea, smart grids through broker [6]).

A priori, we suppose the following:

- Cognitive agents in the typical scenario A demand very simple cognitive and communicative requirements.
- Agents in the typical scenario B need to be modelled using an agent architecture in order to improve the high-demand cognitive abilities. On the other hand, we expect that the medium-demand communicative requirements are satisfied in a very efficient way, since the Jason's communication model is used.
- The high frequency of the cognitive processing and the high complexity of the communicative acts in the typical scenario C (not a simple notifications like in Cardinea) could produce a significant overload of the simulation.

Hence, evidences presented below are obtained over the illustrative typical scenario B, summarizing the main characteristics of Cardinea, where:

1) There are a high number of agents, only few of them need of cognitive abilities.
2) Cognitive agents do not perform cognitive behaviours in all steps of the simulation.
3) Communicative requirements are relatively simple. Basically, dialogues between agents are only based on requesting and notification messages.
4) Additionally, a specific and easy to use model is needed to code the cognitive agent abilities. (This way, encoding and maintenance of agents' knowledge should be simple.)

The evidences are obtained from several simulations, taking into account several numbers of reactive agents (denoted by *n*) and different numbers of cognitive ones (denoted by *nl*). The values of these parameters are set up according to the real requirements of Cardinea. The simulations were repeated 10 times and results averaged after that. The experiments are carried out on a dedicated machine, with GNU/Linux, four Intel(R) Xeon(R) X3230 1.66GHz processors, and 8Gb of memory.

B. Which is the most suitable alternative?

In [15] evidences about preliminary performance evaluations of the two hybrid approaches 2 and 3 (presented above) are collected. These evidences suggest that:

- Considering the alternative 2, the simulation of *n=2500* reactive agents computes very high times (160,32 minutes). There was not possible to obtain evidences for *n=15000* reactive agents (usual number of social agents in several real scenarios like Cardinea), because the memory of JVM was not enough.
- For the same number of reactive agents, the total time of the simulation using alternative 2 are around 100 times the ones produced by alternative 3. Management of the reasoning process, the message interchanging and the underlying agents platform produce a significant overload.
- The total simulation time dedicated to these processes includes (1) reasoning time (consumed by BDI interpreter), (2) time dedicated to interchange messages between cognitive agents (using the underlying agent platform), (3) time dedicated to manage the life-cycle of the agents, basically.

  Using the alternative 2, the major part (more than 99%) of the time is consumed by cognitive processing, the message interchanging and/or the life-cycle management. The difference between simulation times of the two alternatives represents the cost of providing communicative functionalities, since the complexity of the agents is the same for the two alternatives.
- Jason not only consumes more time than MASON to control the simulation, but a lot of time is dedicated to cognitive processing. A significant part of the time could be consumed by (1) the time that agents use to reasoning and (2) the time that underlying agent platform imposes to Jason in order to provide managing and communicative functionalities.

However, a deep evaluation about the performance of two alternatives is required. The first discussion is based on the number of reasoning processes. The figure 1 and the table III compare the two alternatives taking into account the total number of cognitive processes in the simulation and the number of cognitive processes per each timestep, respectively. (Values obtained by using alternative 2 is around 3 times the values obtained by using alternative 3.) They show how the time of the simulation could be reduced by the use of the alternative 3, considering that not all of cognitive agents perform cognitive processes. (Not all of supervisors reason in each timestep, they only perform deliberative and communicative processes when some assistant reaches the limit of contamination.)

On the other hand, the use of alternative 2 causes all cognitive agents perform cognitive processes in each timestep.

TABLE II
COGNITIVE REQUIREMENTS FOR THREE TYPE OF SCENARIOS WHERE SOCIAL AND COGNITIVE AGENTS ARE SIMULATED.

Cognitive requirements	Typical scenario A (e.g. Urban Traffic)	Typical scenario B (e.g. Cardinea)	Typical scenario C (e.g. Smart Brokers)
3) Is an agent architecture required?	No	Yes, could be BDI	Yes, could be BDI
4) Reasoning, planning, explaining requirements	Low	High	High
5) Coordination and communications requirements	Low	Medium	High
6) Frequency of performing complex processing by cognitive agents	Low	Low	High

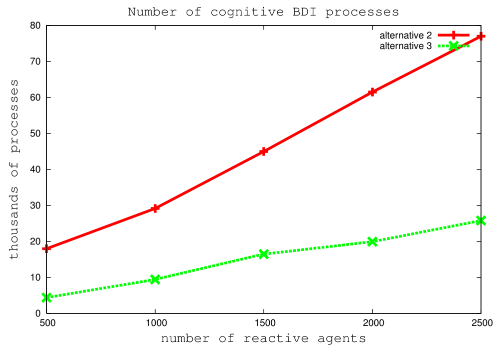

Fig. 1. Total number of cognitive processes carried out during simulation, considering *n* reactive agents and *nl* = *8* cognitive ones.

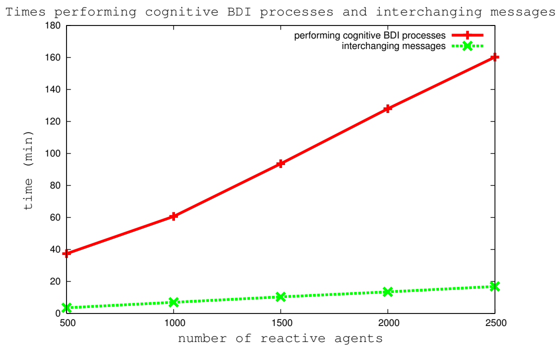

Fig. 2. Time dedicated to performing cognitive processes (including reasoning, message interchanging and life-cycle management) and time dedicated to messages interchanging, for the alternatives 2, considering *n* reactive agents and *nl=8* cognitive ones.

TABLE III
NUMBER OF COGNITIVE PROCESSES, PER EACH SIMULATION TIMESTEP, CONSIDERING *n* REACTIVE AGENTS AND *nl* = *8* COGNITIVE ONES.

Number of reactive agents	n=500	n=1000	n=1500	n=2000	n=2500
Alternative 2	8	8	8	8	8
Alternative 3	2,63	2,73	2,53	2,76	2,69

(The reasoning cycle of each supervisor is activated in each timestep.)

Other significant characteristic of this type of scenario is that the number of communicative acts grow when the number of reactive agents grows too. It is interesting to determine how much time is dedicated to provide communicative functionalities when the number of reactive agents is increased.

The alternative 2 provides communication between agents without any programming efforts, using an underlying agents platform. But, using this alternative the total simulation time is greater than total simulation time of the alternative 3 and the major part (more than 99%) of the time is consumed by cognitive processing, the message interchanging or the life-cycle management (according to the results offered by [15]).

The figure 2 shows that the message interchanging only consumes a few part of this time. For *nl=8* cognitive agents, the time dedicated to the message interchanging is around the 10-12% of the time devoted to reasoning, the message interchanging and the life-cycle managing.

In terms of simulation capabilities, the main difference between alternatives 2 and 3 is the ability of the alternative 2 to provide communicative and coordinative mechanisms based on some underlying agent platform. But, using the alternative 2, the excessive consumption of time is not given by time devoted to the message interchanging. This time is only 10-12% of the time devoted by Jason to manage the simulation and the life-cycle of the agents.

C. Evaluation of alternative 2

The second set of experiments studies how simulation time is consumed in the alternative 2. Times related to the reasoning processes and the message interchanging are measured. The relationships between them are presented in order to determine why this alternative is not suitable to simulate a great amount of reactive agents. (It was not possible to obtain evidences for *n=15000* reactive agents, because the memory of JVM was not enough [15].)

Figure 3 shows the time performing cognitive processes (including reasoning, message interchanging and life-cycle management) and the time dedicated to the message interchanging, for the alternatives 2, for several numbers of cognitive agents *nl=2,3,...10* and two number of reactive ones *n=500,2500*.

For the two numbers of reactive agents, *n=500* and *n=2500*, the time performing cognitive processes increases slightly when the number of cognitive ones increases slightly (from *nl=2* to *nl=10*). The increasing trend is the same for both cases, but in different scales.

Although the time dedicated to the message interchanging increases slightly when the number of cognitive agents increases slightly (from *nl=2* to *nl=10*), it represents a small portion of the time dedicated to perform cognitive processes (see figure 3). Time dedicated to the message interchanging are kept below 15% of the time performing cognitive processes,

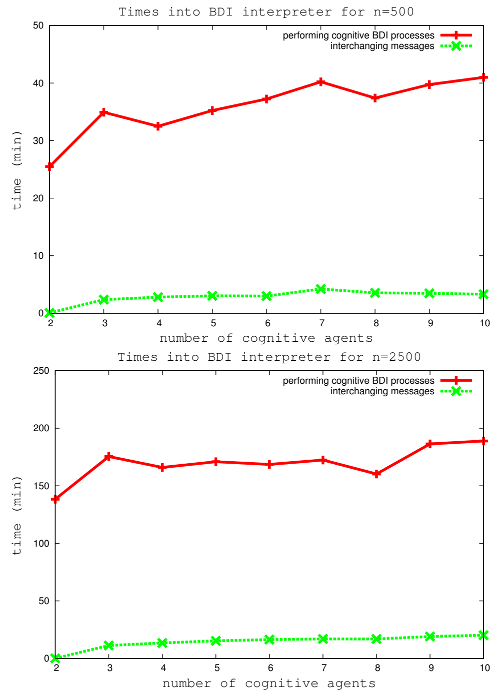

Fig. 3. Relationship between the time performing cognitive processes (including reasoning, message interchanging and life-cycle management) and time dedicated to the message interchanging, for the alternatives 2, considering *nl* cognitive agents and (a) *n=500* and (b) *n=2500* reactive ones.

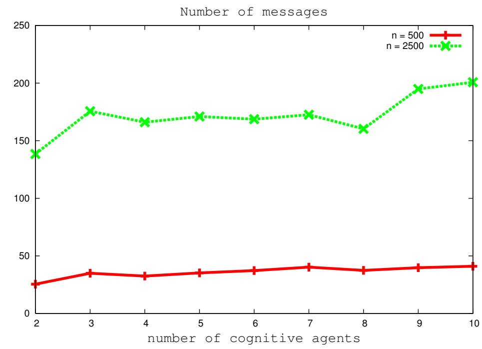

Fig. 4. Total number of messages, interchanged during simulation of the alternatives 2, considering *nl* cognitive agents , and two numbers of reactive ones *n=500, 2500*.

for all numbers of cognitive agents. The relationship between these two times does not seem to depend on the numbers of cognitive agents. The time dedicated to the message interchanging represents a small portion of the time performing cognitive processes, for all considered numbers of cognitive agents (*nl=2, .., 10*). Most of the cognitive processing time is spent on other mechanisms like life- and reasoning-cycle, or any other functionalities or services of the underlying agent platform or the Jason simulation framework. Elucidation about how the time is spent for these mechanisms is out of the scope of this work, and maybe it is only possible if several new breakpoints are considered in Jason simulations.

The time dedicated to the message interchanging increases slightly because the number of interchanged messages increases too. The figure 4 suggests a lineal relation between the number of messages and the number of reactive agents. Also, it shows how the time of the message interchanging varies with the same trend that the number of interchanged messages, for both numbers of reactive agents (*n=500* and *n=2500*). That is because messages are interchanged with the same speed across the system using the message transporting functionalities of the underlying agents platform, independently of the number of reactive and cognitive agents. It means that delay in simulation time it is not produced by limitations in the message transporting system of agent platform and the way that the Jason provides communicative functionalities.

D. Evaluation of alternative 3

Evidences presented in [15] and in section IV-B of this paper demonstrate how alternative 3 is a suitable way to simulate several hundreds (from *n=500* to *n=2500*) of reactive social agents with some of them (*nl=8*) carry out cognitive processes. The behaviours of the cognitive agents, simulated in these experiments, are obtained by a very simple BDI code. Even so, the approach presented by alternative 2 is not able to simulate several thousands of reactive agents.

Evidences offered here try to answer two questions:

1) Is the alternative 3 able to simulate numbers of social agents greater than *n=2500*?
2) Is the alternative 3 able to simulate numbers of cognitive agents different that *nl=8*?

Figure 5 shows the total simulation time and the time performing cognitive processes, for different numbers of agents. Both, the number of reactive agents and the number of cognitive ones are assigned according to the most usual numbers of agents in Ubik or Cardinea scenarios. The experiments consider the most usual values and some greatest ones, in order to explore the scalability of the approach.

Hence, figure 5.a shows the evolution of the total simulation time and the time performing cognitive processes, for different numbers of reactive agents *n=2500,5000,...30000*, considering *nl=8* cognitive agents. On the other hand, figure 5.b offers evidences for several numbers of cognitive agents *nl=2,3,...20*, considering a high number of reactive agents *n=15000*.

On the other hand, this experiment also analyses the number of cognitive process carried out by the two alternatives when the number of agents (both reactive and cognitive) grows. Figure 6 compares the numbers of cognitive processes, for different amounts of agents. The data series of the alternative 2 do not represent values obtained from simulation because this alternative is not able to simulate several hundreds of reactive agents in a finite and short time. The values are estimated using the number of the timesteps when the assistants are kept out the contamination. The final timesteps are the same for the two alternatives and for a particular number of agents. The number of cognitive processes, that alternative 2 should

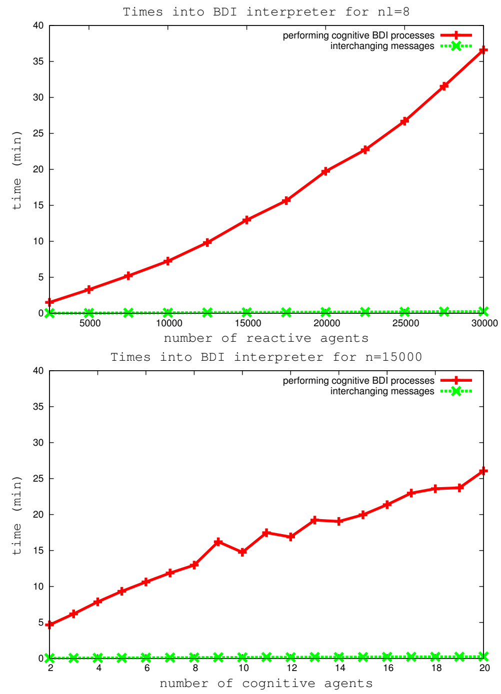

Fig. 5. Relationship between total simulation time and time dedicated to cognitive processes, for the alternatives 3, considering (a) several numbers *n* of reactive agents and *nl=8* cognitive ones, and (b) several numbers *nl* of cognitive agents and *n=15000* reactive ones.

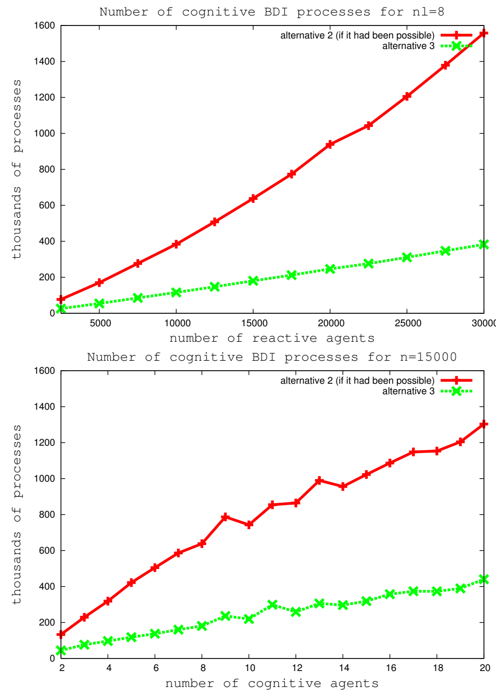

Fig. 6. Number of cognitive processes, for the two alternatives 2 and 3, considering (a) several numbers *n* of reactive agents and *nl=8* cognitive ones, and (b) several numbers *nl* of cognitive agents and *n=15000* reactive ones. (Data presented for alternative 2 are estimated, because this alternative does not support several thousands of reactive agents.).

be carried out, can be estimated because, in each timestep, this alternative performs as many cognitive processes as the number of cognitive agents are simulated.

Figure 6.a compares the number of cognitive processes, for different numbers of reactive agents *n=2500,5000,...30000*, considering *nl=8* cognitive agents. Figure 6.b shows evidences for several numbers of cognitive agents *nl=2,3,...20*, considering a high number of reactive agents *n=15000*. In both cases, the figure suggests that values for alternative 2 are around three times the values for alternative 3.

The alternative 3 is able to simulate a high number of reactive and cognitive agents in a finite and short time. Although the total simulation time increases when the number of reactive agents grows, for *n=30000* reactive agents, the simulation time does not exceed 40 minutes. Also, the alternative 3 is suitable to simulate several cognitive agents in large simulations, considering *n=15000* social agents. Total simulation time does not exceed 30 minutes when *nl=20* cognitive agents are considered. Numbers of cognitive agents greater than *nl=20* was not explored because the number of cognitive agents in several real scenarios like Ubik or Cardinea does not reach this value. However, there is no doubt about the capability of this alternative to support greater values in a society with *n=15000* reactive agents. Exploration of more complex BDI cognitive agents might be interesting in real scenarios.

Time performing cognitive processes does not represent more than 1,35% of the total simulation time in any case.

V. Recommendations for Use

Taking into account the evidences presented in the previous section, several recommendations could be suggested:

1) For typical scenario A (see table II) the use of a standalone social platform like MASON (see the table I) could be the best choice. Hybrid approaches offered by alternatives 2 and 3 also exhibit suitable performances when cognitive requirements of agents are low.

2) The alternative 2 is not suitable for the typical scenarios A and B (see table II) where execution of one reasoning cycle per each timestep is not required. When simulation is controlled by Jason environment (like the alternative 2), all cognitive agents perform one reasoning cycle per each timestep. According to [15], this alternative is able to simulate appropriate numbers of cognitive agents (up to *nl=10*) into social simulation, considering up *n=2500* reactive agents. However, this is not a desired time for several real scenarios.

3) Alternative 2 consumes a lot of time (in order to guarantee communications between agents), the major portion of this time is devoted to reasoning or managing life-cycle of the agents, but not to provide communicative capabilities. Most of the cognitive processing time is spent on other mechanisms like life- and reasoning-cycle, or any other functionalities or services of the underlying architecture of the platform or the Jason simulation framework.

4) The alternative 2 does not support a high number of reactive agents where a few number of them perform cognitive processes (even with a very simple BDI logic description). For instance, using this alternative *n=15000* reactive agents can not be simulated. But, supposing that would be possible, alternative 2 would perform around three times the cognitive processes performed by alternative 3, for all explored configurations.

5) The alternative 3 is suitable to integrate social and cognitive capabilities for agents in the typical scenario B (see table II). It is able to simulate a high number of reactive and cognitive agents in a finite and short time, according to the requirements of the typical scenario C. The portion of the simulation time performing cognitive processes is very small, for all explored configurations where the numbers of reactive and cognitive agents are increased. The time devoted to communicate agents is depreciable in this alternative, because a simple and (time and memory) low-cost mechanism is implemented to communicate agents.

6) Using the alternative 3, small simulation times are obtained since any messages transporting system of underlying agents platform supported by Jason is not used. The characteristics of the messages transporting mechanism, its efficiency, and how it influences the efficiency of the total simulation depend on the requirements of the scenario.

7) This way, incorporating cognitive behaviour in social simulations from the integration of Jason and Mason could not be suitable for typical scenario C, where (1) the frequency of performing cognitive processing and (2) the communicative requirements could be high.

VI. Conclusions and future work

The suitability of several platforms for simulating the most relevant cognitive abilities of the agents in MABSS is examined in this paper. For that, we consider different types of approaches (social, cognitive and hybrids ones). The performance of two hybrid approaches, combining MASON and Jason, are deeply evaluated taking into account the Cardinea scenario, a middle-complexity scenario where agents perform low-frequently cognitive processing and relatively simple communicative acts. Evidences obtained for Cardinea scenario are used to offer several recommendations for use of the hybrid approaches for other types of scenarios. Three types of scenarios with different cognitive requirements are explored in this paper. The best hybrid approach for Cardinea could not be suitable for other scenarios with different cognitive and communicative requirements. This way, a deep evaluation of this approach for more complex agents is required.

Acknowledgment

This work is partially supported by the Spanish national research project TIN2011-28335-C02-02, Foundations for the Development of AAL Services and Applications, through the Bureau of Innovation and Economy.

References

[1] N. Gilbert and K. G. Troitzsch, *Simulation for the Social Scientist. Second edition*. Open university Press. MacGraw Hill Education, 2005.

[2] G. Balan and S. Luke, "History-based traffic control," in *In Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS-2006)*. ACM, 2006, pp. 616–621.

[3] E. Serrano, J. A. Botia, and J. M. Cadenas, "Ubik: a multi-agent based simulator for ubiquitous computing applications," *Journal of Physical Agents*, vol. 3, no. 2, 2009.

[4] CARDINEA, "Documentos de requisitos y casos de uso," Ministerio de Industria, Turismo y Comercio. Avanza I+D 2009. TSI-020302-2009-43, Tech. Rep., 2009.

[5] M. Friedewald, O. D. Costa, Y. Punie, P. Alahuhta, and S. Heinonen, "Perspectives of ambient intelligence in the home environment," *Telemat. Inf.*, vol. 22, no. 3, pp. 221–238, 2005.

[6] P. P. Reddy and M. M. Veloso, "Learned behaviors of multiple autonomous agents in smart grid markets." in *AAAI*, W. Burgard and D. Roth, Eds. AAAI Press, 2011.

[7] S. Luke, C. Cioffi-Revilla, L. Panait, K. Sullivan, and G. Balan, "Mason: A multi-agent simulation environment," *Simulation*, vol. 81, no. 7, pp. 517–527, 2005.

[8] M. Parker, "What is ascape and why should you care?" *Journal of Artificial Societies and Social Simulation*, vol. 4, no. 1, 2001.

[9] J. E. Laird, "Extending the soar cognitive architecture," in *Proceeding of the 2008 conference on Artificial General Intelligence 2008*. Amsterdam, The Netherlands, The Netherlands: IOS Press, 2008, pp. 224–235.

[10] J. R. Anderson, "Act: A simple theory of complex cognition," *American Psychologist*, vol. 51, pp. 355–365, 1996.

[11] M. A. Just and S. Varma, "The organization of thinking: What functional brain imaging reveals about the neuroarchitecture of complex cognition," *Cognitive, Affective and Behavioral Neuroscience*, vol. 3, no. 7, pp. 153–191, 2007.

[12] R. Sun, "Cognitive social simulation incorporating cognitive architectures," *IEEE Intelligent Systems*, vol. 22, no. 5, pp. 33–39, 2007.

[13] M. Okaya, S. Yotsukura, and T. Takahashi, "Robocup 2009," J. Baltes, M. G. Lagoudakis, T. Naruse, and S. S. Ghidary, Eds. Berlin, Heidelberg: Springer-Verlag, 2010, ch. A hybrid agent simulation system of rescue simulation and USARSim simulations from going to fire-escape doors to evacuation to shelters, pp. 414–424.

[14] H. V. D. Parunak, P. Nielsen, S. Brueckner, and R. Alonso, "Hybrid multi-agent systems: Integrating swarming and bdi agents," *Engineering SelfOrganising Systems*, vol. 4335/2007, no. Lecture Notes in Computer Science, pp. 1 – 14, 2007.

[15] A. Caballero, J. Botía, and A. Gómez-Skarmeta, "Using cognitive agents in social simulations," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 7, pp. 1098–1109, 2011.

[16] M. North, N. Collier, and J. Vos, "Experiences creating three implementations of the repast agent modeling toolkit," *ACM Transactions on Modeling and Computer Simulation*, vol. 16, no. 1, 2006.

[17] F. Klügl, R. Herrler, and M. Fehler, "Sesam: implementation of agent-based simulation using visual programming," in *AAMAS '06: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*. New York, NY, USA: ACM, 2006, pp. 1439–1440.

[18] R. H. Bordini and J. F. Hübner, *Adelinde M. Uhrmacher and Danny Weyns (Eds.) Agents, Simulation and Applications*. CRC Press Inc, Taylor Francis Group: Boca Raton London New York, 2009, ch. Agent-Based Simulation Using BDI Programming in Jason.

[19] C. Nikolai and G. Madey, "Tools of the trade: A survey of various agent based modeling platforms," *Journal of Artificial Societies and Social Simulation*, vol. 12, no. 2, 2008.

[20] R. Allan, "Survey of agent based modelling and simulation tools," Computational Science and Engineering Department, STFC Daresbury Laboratory, Tech. Rep., 2009.

[21] R. H. Bordini, J. F. Hübner, and M. Wooldridge, *Programming Multi-Agent Systems in AgentSpeak using Jason*. Addison-Wesley, 2007.

[22] K. Hindriks, F. de Boer, W. van der Hoek, and J.-J. Meyer, "Agent programming in 3apl," *Autonomous Agents and Multi-Agent Systems*, vol. 2, no. 4, pp. 357–401, 1999.

[23] R. Evertsz, F. E. Ritter, P. Busetta, and M. Pedrotti, "Realistic behaviour variation in a bdi-based cognitive architecture," in *In Proceedings of SimTecT 2008*. Melbourne, Australia: SIAA Ltd., 2008.

[24] A. S. Rao, "Agentspeak(l): Bdi agents speak out in a logical computable language," in *Walter Van de Velde and John Perram, editors, Proceedings of the Seventh Workshop on Modelling Autonomous Agents in a Multi-Agent World (MAAMAW96)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1996, pp. 42–55.