

# Connecting Game Theory and Multi-Agent Systems

Daniel L. Kovacs<sup>a)</sup>

Department of Measurement and Information Systems, Budapest University of Technology and Economics, P.O. box 91, H-1521 Budapest, Hungary, Tel/Fax: +36-30-301-7886<sup>a)</sup>  
e-mail: dkovacs@mit.bme.hu

## ABSTRACT

The problem of designing a given collective behavior (e.g. cooperation, negotiation, altruism) in a Multi-Agent System (MAS) is a well known issue, still there is no general approach to solve it. In fact, there is no general and comprehensive theory relating the individual behavior of agents with the collective behavior of the MAS, which would enable uniform analysis and synthesis of such systems in general. The current article proposes a new approach to this problem based mainly on game theory.

## INTRODUCTION

Multi-Agent Systems (MAS's) model distributed systems with a collective of autonomous agents embedded in an environment. An *agent* "can be anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors". [1] The problem of designing a given collective behavior in a MAS is still an unsolved issue in general. Nonetheless there are theories, which capture some profound aspects of the problem (e.g. distributed AI [2], game theory [3], theory of implementation [4], satisficing games [5]). These theories are mostly analytical, and differ in their level of abstraction. By combining them it would be possible to design and analyze MAS's in a uniform, general way from the detailed low level representation of single agents up to the high level representation of the collective. Important properties of the system (rationality, efficiency, etc) could be examined in general. Also universal criteria (resource consumption, optimality, etc) could be satisfied formally.

As promising as it seems, it has also its pitfalls. The different representations of MAS's differ fundamentally. They differ in:

- *origin*: e.g. game theory is an economical theory, while distributed AI is part of computational science;
- *assumptions*: e.g. theory of implementation assumes agents to act according to some kind of game theoretical solution concept, while distributed AI models their decision mechanism explicitly; the former assumes common

knowledge of the agent-environment, while the latter usually doesn't;

- *formalism*: e.g. game theory proposes a general mathematical framework, while distributed AI proposes many different, even ad-hoc representations (e.g. [6], [7]);
- *level of abstraction*: e.g. game theory doesn't consider the physical aspects of the agent-environment, while PDDL [8] focuses mainly on that, leaving behind the issue of representing the high level strategic interaction among agents;
- *implementability*: e.g. game theory is descriptive, i.e. it is used primarily for **analysis**, while distributed AI aims at the design of groups of agents, i.e. **synthesis**.

All these issues need to be covered to bridge the gap between the different representations reducing them to a common denominator. In the following a solution is proposed: a new concept based on game theory. After that, a MATLAB-based implementation is outlined, followed by the conclusions.

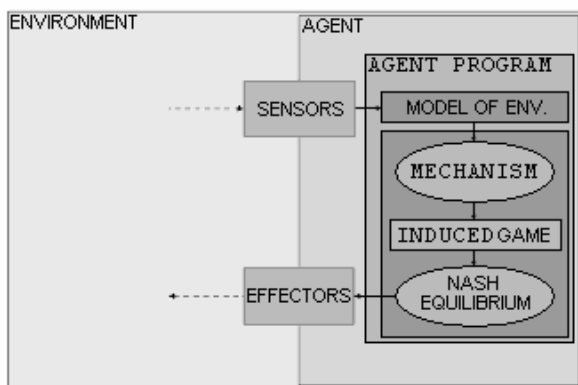
## CONNECTING GAME THEORY AND MULTI-AGENT SYSTEMS

Connecting game theory and MAS's is not a new topic. In the last decade game theory became increasingly popular among distributed AI researchers. It is used mostly for analysis of special MAS tasks (e.g. communication [9]), but there is still no general unification of the two approaches.

Game theory and MAS's address the same issue at different levels of abstraction. The former considers the agent-environment without representing its physical aspects nor the structure of the agents, while the latter concentrates mainly on the rich details of the individuals. A connection follows intuitively.

From the point of view of an external observer an agent-environment in general is just an *environment with incomplete information* [4], where agents are *players*; agents' conditional plans are *strategies* of the players; agents' possible architectures for running agent-programs [10] are the possible *types* of the players; agents' beliefs correspond to the *probability distributions* over these types; and a collective behavior in the MAS is a *strategy-combination* of the players.

Now the **program of an agent** can be described in the following way: it *models* the agent-environment and then – based upon that model – *decides* what action/plan to execute. This *decision mechanism* can also be described in a game theoretical way: there is a *mechanism* [4] imposed upon the model of the environment, which induces a Bayesian game [11], from which a Nash-equilibrium is selected. A *Nash-equilibrium* is a strategy-combination, where no agent has the incentive to change its strategy unilaterally. Agents act according to the strategy prescribed (to their self-representation) by that equilibrium. Fig. 1 illustrates the concept.



**Fig. 1** The concept of connecting Game Theory and Multi-Agent Systems.

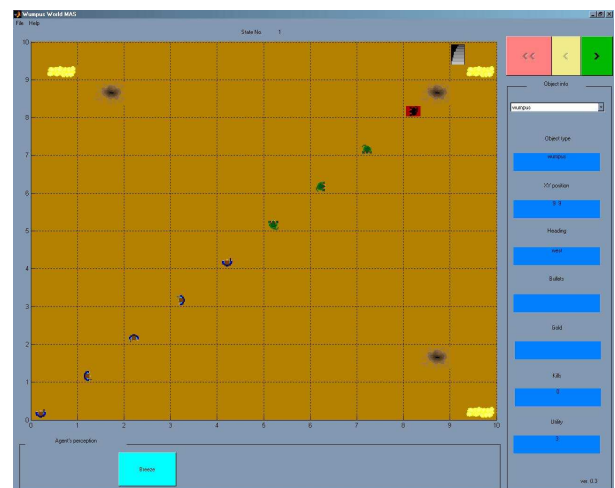
Lately it was formally proven, that this way any collective behavior in a general MAS can be implemented exactly in the sense of implementation theory [12].

### REALIZATION OF THE CONCEPT

The concept was realized in MATLAB programming environment for the multi-agent version of Wumpus World (WW-MAS) [1]. It makes an excellent testbed for intelligent agents. The test results supported the main theoretical statement: any collective behavior – given by a Social Choice Function (SCF) [4] – could be achieved even in the case when every agent shared the same model of the environment. I.e. if the necessary collective behavior was given, then in every test-case the agents acted as specified. Fig. 2 shows a screenshot of this test-application.

### CONCLUSIONS

The article presented a novel method to specification facilitating design and analysis of collective behavior in MAS's. This way arbitrary collective behavior can be implemented exactly in general. Future research will mainly focus on synthesis: connecting the concept with different low-level representations (e.g. PDDL).



**Fig. 2** A screenshot of the MATLAB implementation of the WW-MAS in the case of a 10\*10 grid, 5 goldminer agents, 4 wumpus agents, 3 pieces of gold, 3 pits, and 1 exit at the top right corner of the grid.

### REFERENCES

- [1] S. Russell, P. Norvig, *Artificial Intelligence: A Modern Approach*, Prentice Hall (2003)
- [2] G. M. P. O'Hare, N. R. Jennings, eds., *Foundation of Distributed Artificial Intelligence*, Wiley (1994)
- [3] J. von Neumann, O. Morgenstern, *Theory of games and economic behavior*, Princeton University Press (1947)
- [4] R. Serrano, "The Theory of Implementation of Social Choice Rules," *SIAM Review*, **46**, 377–414 (2004)
- [5] W. C. Stirling, M. A. Goodrich, "Satisficing games," *Inf. Sci.*, **114**, 255–280, (1999)
- [6] P. O'Brien, R. Nicol, "FIPA - Towards a Standard for Software Agents", *BT Tech. Journ.*, **16:3**, 51–59 (1998)
- [7] L. Gulyás, T. Kozsik, J. B. Corliss: „The Multi-Agent Modeling Language and The Model Design Interface”, *Journ. of Art. Soc. and Soc. Sim.*, **2:3** (1999)
- [8] D. McDermott and AIPS'98 IPC Committee. PDDL—the planning domain definition language. Tech. rep. (1998)
- [9] P. J. Gmytrasiewicz, E. H. Durfee, *Rational communication in multi-agent environments*. Aut. Agents and Multi-Agent Sys., **4**, 233–272 (2001)
- [10] S. Russell, D. Subramanian, "Provably bounded-optimal agents," *Journ. of AI Res.*, **2**, 1–36 (1995)
- [11] J. C. Harsányi, „Games with incomplete information played by Bayesian players I-II-III,” *Man. Sci.*, **14**, 159–182, 320–334, 486–502 (1968)
- [12] D. L. Kovács, "Virtual Games: A New Approach to Implementation of Social Choice Rules," *LNCS*, **3690**, 266–275 (2005)