Multi-Agent Coordination is Arbitration from a Super-Holon Point of View

Sebastian Rodriguez, Vincent Hilaire, and Abder Koukam

Systems and Transportation laboratory
UTBM
F-90010 Belfort Cedex, FRANCE
{sebastian.rodriguez,vincent.hilaire,abder.koukam}@utbm.fr
+33 384 583 009

Abstract. This paper proposes an approach for engineer self-organised Multi-Agents Systems. This approach is based upon the human immune system. The role of the immune systems, in this approach, is the arbitration of several behaviours. We extend the arbitration capabilities of the immune system to define an architecture for multi-agent systems coordination. The proposed architecture is composed of several levels. Each level corresponds to an immune system. The behaviour resulting from the arbitration of a level is used as input by the level immediately below. This architecture allows the coordination of interacting entities at several level of abstraction. We illustrate this approach with a MAS dedicated to robot soccers simulation.

1 Introduction

Complex systems are often characterized by networks of numerous interactive entities. They are called complex because of the complexity of the exhibited behaviours. These behaviors are the result of the non linear aggregation of the local behaviours of their components. Multi-Agent Systems have become a natural tool for modeling, simulating and programming complex systems. Indeed, Multi-Agents Systems are composed of autonomous, reactive, proactive and interacting entities called agents engaged in the realisation of a joint goal. Both type of systems are notably studied by their organisation dynamics and by the emergence of organisational structures. The engineering of self-organised MAS is a hard problem. Many mechanisms have been proposed [20]. Among these mechanisms some are inspired by nature or biological phenomena, such as Ant Colony Systems [8]. The goal of this paper is to present such a mechanism inspired by the human immune system. We have designed a MAS architecture based upon the immune system which enables self-organisation and emergence of coordinated organisational structures.
Coordination of agents within a Multi-Agent Systems (MAS in the sequel) has long been discussed in the MAS community [25]. It is a hard problem. Indeed, agents are autonomous entities interacting in open dynamic environments to achieve their goals. Agents may have different goals and a group of agents may

have a joint goal. Numerous architectures have been proposed in order to coordinate agent's behaviors towards the accomplishment of system goals.
The approach proposed in this paper is based upon the use of an arbitration mechanism extended by using several levels to deal with the multi-agent case.
The arbitration mechanism chosen is inspired by the human immune system. This metaphor considers stimulations/requests as antigens and selects antibodies as reactions/answers. This mechanism allows an agent to choose among a set of possible behaviours the one which seems the fittest for a specific context. Each agent has thus a set of antibodies, representing behaviours, fitted to respond to specific antigens, representing stimulus.
In order to coordinate a group of agents engaged in a joint goal our approach proposes to see the group as an agent or a holon referring to the holonic paradigm [11]. Defined by Koestler [16], holons are self-similar entities that can not be considered as wholes nor parts in an absolute sense. According to Koestler, a holon is a self-similar structure that consists of several holons as sub-structures. The behavior of the group is seen from the outside as the behavior of a single agent/holon. We propose to use an immune system for this holon that selects an antibody. This antibody represents the behavior to be executed by the members of the holon. The chosen behavior acts as input for the sub-holons. Each sub-holon takes the behaviour as a constraint and reacts with it own immune system to the perceived antigens.
The advantages of this approach are numerous. Each agent keeps its autonomy but coordination towards goal accomplishment is ensured by selecting the right stimulus. A MAS may pursue several goals at the same time. These goals may be interdependants. In addition, the immune system allows learning by a reinforcement mechanism.
The natural immune system is a subject of great research interest because it provides powerful and flexible information processing capability as a decentralised intelligent system. The immune system provides an excellent model of adaptive operation at the local level and of emergent behavior at the global level. There exists several theories to explain immunological phenomena and software models to simulate various components in the immune system (IS) [23].
The basic components of the immune system are macrophages, antibodies and lymphocytes. Lymphocytes are the cells maturing in the bone marrow and producing antibodies from its surface. The antibody recognizes and eliminates a specific type of antigens (foreign substances invading a human body). The key portion of antigen that is recognized by the antibody is called epitope, which is the antigen determinant. Paratope is the portion of antibody that corresponds to a specific type of antigen. Once an antibody combines an antigen via their epitope and paratope, the antibody starts to eliminate the antigen. Recent studies in immunology have clarified that each type of antibody also has its own antigenic determinant, called an idiotope. This means an antibody is recognized as an antigen by another antibodies [9]. Based on this fact, Jerne proposed the concept of the immune network, or idiotypic network [14], which states that antibodies and lymphocytes are not isolated, but they are communicating with each

other. The idiotope of an antibody is recognized by another antibody as an antigen. This network, called idiotypic network, is formed on the basis of idiotope recognition with the stimulation and suppression chains among antibodies. The approach presented in this paper is based on this interpretation of the immune system.
This paper is organised as follows: section 2 presents the principles of the idiotypic network. Section 3 presents the concepts of the multi-level idiotypic network architecture and illustrates it on the robot soccer example. Eventually, section 4 concludes.

2 Immune System Based Agent Architecture

The Jerne's idiotypic network has already been used as agent architecture in [24]. We use the concepts developed in this reference for a single agent case as a basis for the approach presented in this paper. First, there is an analogy between antibodies and agent's concepts. This analogy is depicted in the figure 1.

precondition under which this B-agent is stimulated	attributes, codes, data, behavior and intern idiotypic network	references to stimulating B-agents and the degree of the stimuli (affinity)
Paratope	Agent specification	Idiotope

Fig. 1. Antibody description

An antibody is divided in three parts. The first part is the precondition. It states under which circumstances the antibody is stimulated. That is to say in which context this antibody may execute it's associated behavior. The precondition is an analogy with the real antibody paratope which tries to match antigens epitope in order to recognize them.
The second part specifies the behavior of the antibody. It is the behavior which is executed when the antibody is selected. The behavior of the real antibody consists in eliminating antigens.
The third part is composed of references to other antibodies and a degree of stimulation or inhibition (affinity). It is the idiotope part of the antibody by which it is recognized and which allows interactions (stimulation/suppression) with other antibodies.
The Jerne's Idiotypic network is defined by the different antibodies and their affinities. The affinities are either stimulation or suppression between two antibodies. An example of idiotypic network is presented in the figure 2.

The i-th antibody stimulates M antibodies and suppresses N antibodies. m_{ji} and m_{ik} denote affinities between antibody j and i, and between antibody i and k, respectively. The affinity means the degree of stimulation or suppression. m_i is an affinity between an antigen and antibody i. The antibody population

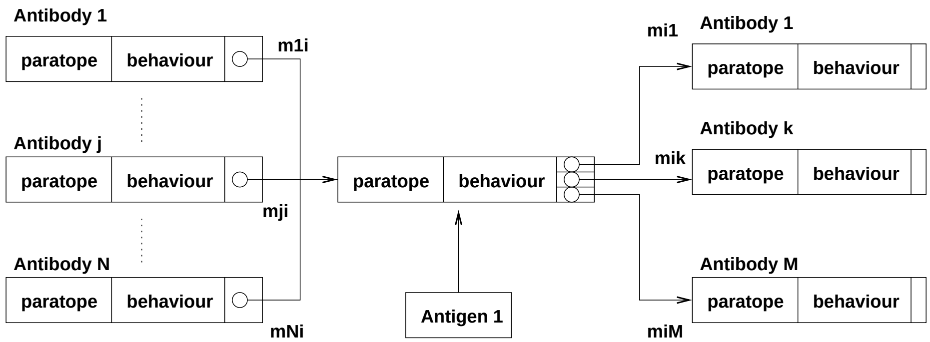

Fig. 2. Idiotypic network

is represented by the concept of concentration. The concentration of the i-th antibody, denoted by a_i, is calculated with the following equations [9]:

$$\frac{dA_i(t)}{dt} = (\alpha \frac{1}{N} \sum_{j=1}^{N} m_{ji} a_j(t) - \alpha \frac{1}{M} \sum_{k=1}^{M} m_{ik} a_k(t) + \beta m_i - k_i) a_i(t) \tag{1}$$

$$a_i(t) = \frac{1}{1 + exp(0.5 - A_i(t))} \tag{2}$$

In the first equation, the first and second term of the right hand side denote the stimulation and suppression from other antibodies. m_{ji} and m_{ik} are positive values between 0 and 1. The third term, m_i, is 1 when antibody i is stimulated directly by an antigen, otherwise 0. The forth term, k_i, denotes the dissipation factor representing the antibody's natural death. The second equation is the function that is used to squash the parameter $A_i(t+1)$, calculated by the first equation, between 0 and 1. An antibody is selected based on the probability proportional to the current concentrations (roulette-wheel selection strategy).
The adaptation mechanism is usually classified into two types: adjustment on-line mechanism and innovation mechanism [**?**]. This article only takes in consideration the on-line adjustment mechanism that initially starts with no affinity between antibodies. This means that the idiotopes of the antibodies are undefined, and obtains the idiotopes using reinforcement signals so the idiotypic network structure can be rearranged at run-time by changing affinity values.
It means that the affinity values are all initialized with 0. The adaptation consists in computing the affinity values using the analysis of reinforcement for executed antibodies behaviors. This process constructs the idiotypic network.

The following equation describes how affinity m_{12} between antibody 1 and antibody 2 is computed.

$$m_{12} = \frac{T_p^{Ab1} + T_r^{Ab2}}{T_{Ab1}^{Ab2}}$$

- T_p^{Ab1} is the number of times penalty reinforcement signals were received when $Ab1$ was selected.
- T_r^{Ab2} is the number of times reward reinforcement signals were received when $Ab2$ was selected.
- T_{Ab1}^{Ab2} is the number of times both, $Ab1$ and $Ab2$, have reacted to specific antigens.

This means that immune network learns from results of its own behaviors.

3 Multi-Level Architecture

3.1 Concepts

The immune system based architecture presented in the precedent section allows the arbitration among several behaviors. The arbitration is the result of the selection of the antibody with the highest concentration. The antibodies concentrations evolve according to stimulation of antigens (perception) and stimulation/inhibition of other antibodies. This architecture doesn't tackle the coordination of interacting agents. Indeed, an agent encapsulates an immune system which chooses the behavior it executes to react to external stimulations. In order to design MAS with coordination capabilities we propose to use several levels of immune systems.

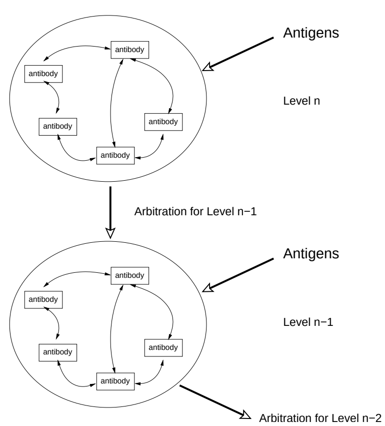

Fig. 3. Two adjascent abstract levels

By using several levels we adopt an holonic point of view [3,11,19]. Indeed, each level arbitrates the behaviors of an entity. A level n entity can be decomposed in sub-entities, of level n-1, with their own behaviors. The level n entity is

a super-holon and entities of level n-1 are its sub-holons. Another viewpoint is to consider the group of n-1 level entities as a whole which is the n level entity or the super-holon. The sub-holons as a group are coordinated by the behavior selected by their super-holon. This choice is the result of the upper level immune system and is taken as input by the lower level immune systems as depicted in figure 3. According to the choice of their super-holon and their perceptions each sub-holon using its own immune system selects a behavior to execute.
The advantages of such a approach are threefold. First, it enables the decomposition of behaviors at several levels of abstraction. This decomposition makes them simpler to design. Second, it separates the different goals of the system and their learning mechanisms. Finally, it allows the coordination of agents and dynamic self-organisation of the MAS.

3.2 Robot soccer team example

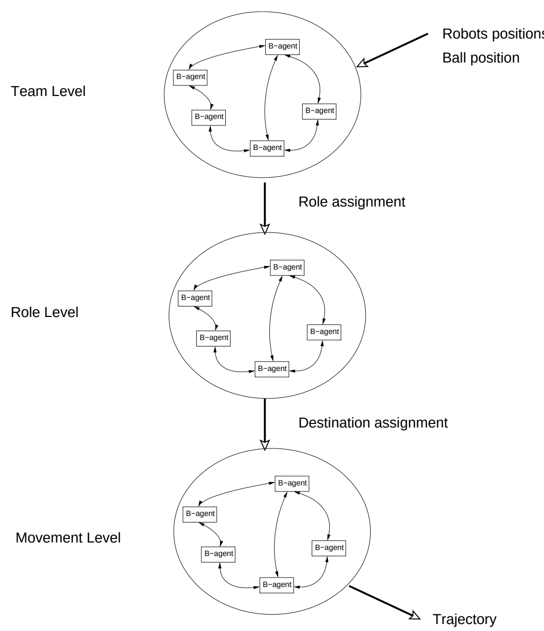

Fig. 4. Overview of the immune systems organisation

The FIRA Robot soccer competitions began in 1996 using real robots and simulators [15]. It is an example where real-time coordination is needed. Indeed, the principle is for two teams of autonomous robots to play a game similar to human football. It constitutes a benchmark for several research fields, such as MAS, image processing and control.

Taking the FIRA contest as case study, we have developed a three layer system to coordinate a robot soccer team. Each layer deals with a different level of abstraction, as shown in figure 4

First, the high level immune system, called team level, assigns appropriate roles to robots according to the situation. These roles are defined by the strategy followed by the team as a whole (defense, attack, etc). This immune system is placed in each team agent.

Second, the medium level immune system deals with role-playing by robots. It computes an aim point for the robot corresponding to the role decided by the high level immune system.

Third, the low level immune system computes the robots trajectories.

As described above, three immune networks have been designed in order to implement strategies for robot soccer. For the basic (primitive) skills such as movements to fixed positions and collision avoidance, the low level idiotypic network was created. Since each robot has information about its environment, it is possible to detect target's positions and obstacles. In this study, the low level immune system's antigen represents either the aim point (generated by the medium level IS)or obstacles (such as others robots and walls). For simplicity the directions are categorized as: front, right, left, back. The sets of preconditions are defined as follows: { *ObstacleFront* , *ObstacleLeft* , *ObstacleRight* , *ObstacleBack* , *None* } and {*AimFront* , *AimLeft* , *AimRight, AimBack* } . The behaviors corresponding to the $ObstacleFront$ precondition are { *MoveLeft* , *MoveRight* , *MoveBack* } . An example of antibody could then be :

Precondition	Behavior	Affinities
ObstacleFront	MoveRight	

The affinities part is empty as it is constructed when the system evolves. These affinities are computed using the reinforcement mechanism. Each robot receives reward signals when current distance to aim (measured by hamming metric) is lower than in previous step. It receives penalty signals either when it collides with an obstacle or when the distance to the aim is greater than in the previous step. It is possible to easily add or remove antibodies from this immune network in order to improve or limit robot's behavior.

The lower level immune system allows robots to move to fixed points. In order to introduce more sophisticated behaviors a second immune system is defined.
This second immune system is called the medium level immune network. Medium level network takes as input: (1) a role produced by the high level immune network and (2) the game context. The different roles define general principles of a behavior. For example, the purpose of the *goal-classic* role is to keep opponent robots from scoring goals. The defined roles act as preconditions for the antibodies. It means that for each role there may be several antibodies. The behaviour of each antibody consists in a list of displacements. Indeed, in the robot soccer game the robots don't have any actuator. The only way of acting is to move in order either to push the ball or block an opponent. We have thus chosen to

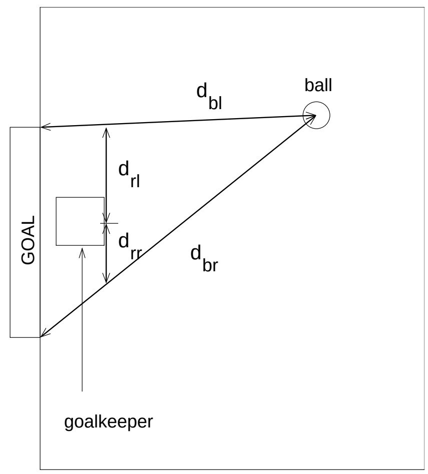

Fig. 5. Goalkeeper reinforcement criterion

represent roles by lists of displacements. Each list contains several fixed points representing the displacement trajectory. Each point of the chosen vector is given as input to the lower level immune system. The approach chosen instead of predefining antibodies as in the lower level immune system is to generate a fixed number of antibodies. The displacement vector of each antibody is generated randomly. An example is shown in figure 6. For the *defense-far* role there are three antibodies representing three different trajectories. If the *defense-far* role is assigned to this robot one of these trajectories will be selected by the medium level immune system. The role of the robot will then consists in moving to the fixed points of the trajectory. Several displacement vectors are generated for each

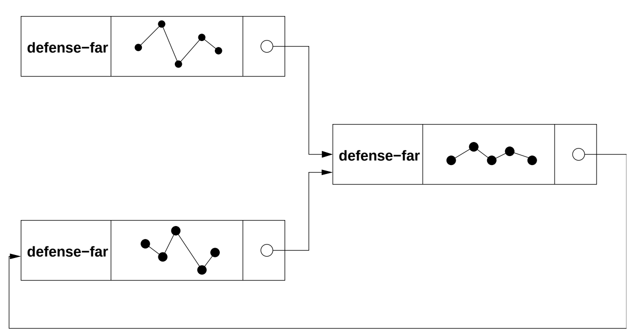

Fig. 6. Examples of medium level antibodies

role and medium level network choose only one. The result of the corresponding antibody is analyzed after it execution. Fitness of chosen action must be defined in order to confirm validity of reinforcement mechanism and is role dependant. For instance in the case of the $goal - classic$ role the criterion depicted in figure 5 was assumed. The idea is to maintain $\frac{d_{rl}}{d_{bl}} = \frac{d_{rr}}{d_{br}}$ relationship. If this relationship is met then the selected antibody receives reward reinforcement signal, otherwise the antibody receives penalty signal. Each robot should be considered

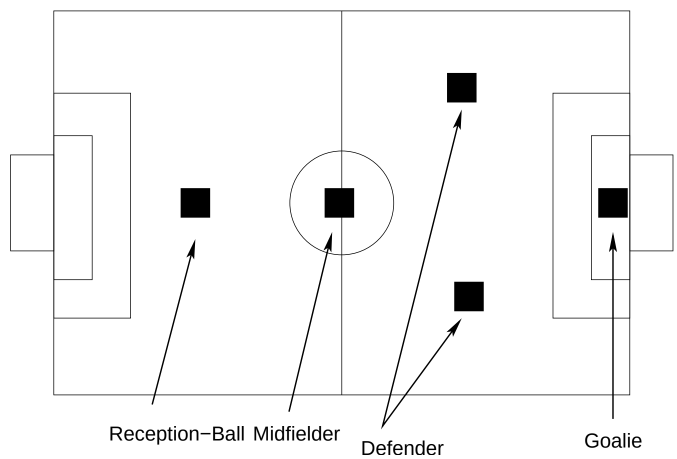

Fig. 7. Example of aim points

as a fully functional soccer player, so different roles were designed. Each role is part of a strategy and corresponds to a specific behavior. Strategies may be composed by different sets of roles. According to the chosen concrete strategy a fixed number of roles must be defined. For instance, one of the several defense strategies for a five players team, defines the following set of roles: { *goal-classic, defense-far, defense-close, support-defense, reception-ball* } . Figure 7 describes the aim point corresponding to a specific strategy. For simplicity reasons aim points are predefined but they can easily depend on game situation or other parameters. According to the current strategy the team assigns a particular role to each robot.

The high level network was designed to coordinate robots for a robot soccers environment. It means that the network has to choose a fitted strategy and assigns roles defined by the strategy to robots. The strategy choice depends on the game situation. This situation is defined by several parameters which act as precondition for antibodies. The first parameter is which team controls the ball. It is determined by the robot which is nearest to the ball. The second is in which zone of the field is the ball. To reduce the set of preconditions the ground has been splitted into 9 equal zones.

According to the current situation and the chosen strategy the high level immune network assigns roles to robots. As was described above each robot can play its own role as, for example, attacker, goal keeper or defender. The concept of zone can be used to assign the appropriate role to each robot.
The fitness of each strategy has to be evaluated. If we sort strategies in two categories : defense and attack. One criterion could be that no goal was taken and respectively a goal was scored. This criterion is assumed as long as the antibody representing the strategy is active.

3.3 Results

We have developed a robot soccer simulator and an immune system API using the MadKit platform [12]. The interface of this simulator during a game is shown in figure 8. This simulator enables us to test different immune networks notably those described in the precedent section.

Fig. 8. Robot Soccers Simulator

The tests we have run were made with two teams of five robots playing against each other. Each team used the same mechanism we have described in this paper. At the beginning of the game the immune networks are initialized without any affinities.
The first phenomenon we have observed is the capacity of the immune systems to adapt and to learn how to play. We have measured the time between two scored goals. This evolution is shown in the figure 9. The discretized units of x

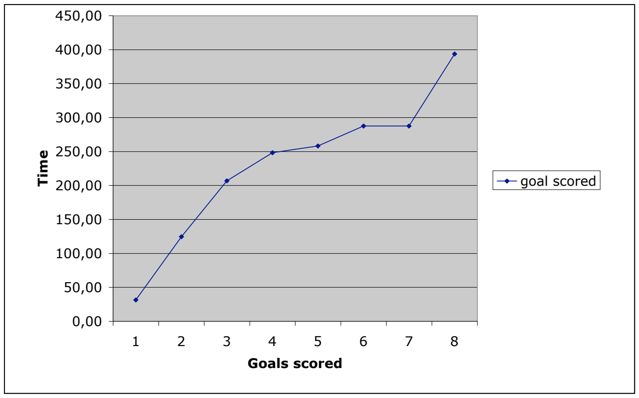

Fig. 9. Time evolution between two goals scored

axis represents the scored goals. The y axis represents the time in second. One can see that the tendency is that time between two goals become more and more important. As the immune networks evolve each team play more efficiently and it becomes more difficult to score a goal.
The second phenomenon we have studied is how each team learned to coordinate its members. More precisely, we have studied the evolution of strategy choices during the games. This evolution is shown in figure 10. The discretized units of x axis represents the simulation steps. The y axis represents the different strategies the high level immune network arbitrates. We have limited this number to four for readability reasons.
At the beginning of the game teams change their strategy often as affinities are in construction and there are no real link between strategies. As the game lasts, strategy changes occur less often.

4 Related Works and Conclusion

Well-known arbitration architectures, such as subsumption of Brooks [2] and Agent Network Architecture (ANA) of Maes [17], don't take into account the coordination aspects in MAS. Moreover, subsumption is based upon a fixed hierarchy of modules which may be hard to design for any application and doesn't allow flexible behaviors. In ANA the modules aren't sorted but they are predefined.
It is indeed difficult to find a MAS architecture which can fit in with dynamic environments and open systems. Self-organization is an answer to such prob-

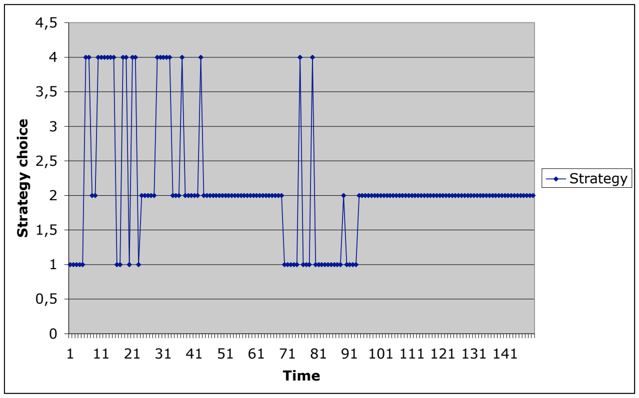

Fig. 10. Strategy choice

lems. Coordination in self-organising systems [20] can be achieved by several techniques such as coordination fields [4] or "tuples on the air" like techniques which enable pheromone propagation [18]. However, these techniques are aimed at very simple behaviors such as exploration. Moreover, it is difficult to take into account multiple objectives.

The metaphor of the immune system has already been used in computer science for a wide variety of applications, e.g. [1,5,10]. For surveys on immune systems applications one can read [6,7]. There are several computational models used to implement interpretations of the immune system. Among them: the idiotypic network of Jerne, used for example in [26,22,1], and the clonal selection mechanism [13]. The latter is based upon the proliferation of antibodies that best match detected antigens. Many applications of immune system deal with recognition of foreign intrusions to distinguish self and non-self entities. This recognition ability has been applied to the computer security field, for example [10,21]. Our work is concerned with immune systems as arbitration mechanism. So behaviors must be associated with antibodies. Examples of such use of the immune system are presented in [26,22]. To our knowledge, among these approaches, none tackles the problem of multi-agent coordination. Moreover, system goals are often unique instead of our approach which allow an objective by level.

In this paper we have presented an approach for the engineering of self-organised MAS based upon several levels of immune systems. The role of the immune systems is the arbitration of several behaviours. The behavior resulting from the arbitration of a level is used as input by the level immediately below. This architecture allows the coordination of interacting entities at several level of abstraction. We have illustrated this approach with a MAS dedicated to robot soccers simulation.

The novelty of the model comes from the use of multiple (hierarchically composed) idiotipic networks. The approach enables a flexible and adaptive behavior without the need of a centralized control. Even more, from a Software Engineering point of view, it reduces the complexity of designing such systems by tackling different levels of abstraction (team/group, robot / individual, trajectory) independently. The advantages of this approach are numerous. The immune system has learning capabilities. It means that it is able to find at runtime the appropriate behaviors to different stimuli. It enables the decomposition of behaviors at several level of abstraction making them simpler to design. It separates the different goals of the system and their learning mechanisms. This goal modularity makes the design easier and the adaptation more pertinent. Eventually, it allows the coordination of agents and dynamic self-organisation of the MAS. Indeed, each holon behaviors help its sub-holons to react to stimulations and to organise in order to solve the problem at hand.

Future research directions include the study of other types of antibodies and antigens in each level in order to test the systems responses with a greater number of possibilities. We will also apply this coordination mechanism to other applications.

References

1. M. Bakhouya, J. Gaber, and A. Koukam. Adaptive service discovery and composition in ubiquitous computing. In *International Conference on Information and Communication Technologies: from Theory to Applications*, 2004.
2. R. Brooks and J. H. Connell. Asynchronous distributed control systems for a mobile robot. *SPIE*, 727, 1986. Mobile Robots.
3. H. Van Brussel, J. Wyns, P. Valckenaers, L. Bongaerts, and P. Peeters. Reference architecture for holonic manufacturing systems: Prosa, 1998.
4. Jrôme Chapelle, Olivier Simonin, and Jacques Ferber. How situated agents can learn to cooperate by monitoring their neighbors' satisfaction. In *15th European Conference on Artificial Intelligence*, 2002.
5. D. Dasgupta and S. Forrest. Tool breakage detection in milling operations using a negative-selection algorithm, 1995.
6. Dipankar Dasgupta and Nii Attoh-Okine. Immunity-based systems: A survey. In *Proc. of the IEEE Int. Conf. on Systems, Man and Cybernetics*, Piscataway, NJ, 1997. IEEE.
7. Leandro Nunes de Castro and Fernando Jos Von Zuben. Artificial immune systems: Part ii - a survey of applications. Technical Report DCA-RT 02/00, Department of Computer Engineering and Industrial Automation, School of Electrical and Computer Engineering, State University of Campinas, SP, Brazil, February 2000.
8. Marco Dorigo and Luca Maria Cambardella. Ant colony system: A cooperative approach to the traveling salesman problem. *IEEE transactions on Evolutionary Computation*, 1(1), 1997.
9. J. Doyne Farmer, Norman H. Packard, and Alan S. Perelson. The immune system, adaption and machine learning. *Physica D*, 22:187–204, 1986.

10. Noria Foukia. IDReAM: Intrusion detection and response executed with agent mobility. In *Engineering Self-Organising Systems*, pages 227–239, 2004.
11. Christian Gerber, Jörg Siekmann, and Gero Vierke. Holonic multi-agent systems. Research Report RR-99-03, DFKI, 1999.
12. Olivier Gutknecht and Jacques Ferber. Madkit : A generic multi-agent platform. In Carles Sierra, Maria Gini, and Jeffrey S. Rosenschein, editors, *Proceedings of the Fourth International Conference on Autonomous Agents*, pages 78–79, Barcelona, Catalonia, Spain, June 2000. ACM Press.
13. Ron Hightower, Stephanie Forrest, and Alan S. Perelson. The Baldwin effect in the immune system: Learning by somatic hypermutation. In Richard K. Belew and Melanie Mitchell, editors, *Adaptive Individuals in Evolving Populations: Models and Algorithms*, pages 159–167. Addison Wesley, Reading, MA, 1996.
14. N. K. Jerne. Towards a network theory of the immune system. *Ann Immunol (Inst Pasteur)*, 125C:373–389, 1974.
15. Yung Ho Kim. *Micro-robot world cup soccer tournament*. KAIST, 1996.
16. Arthur Koestler. *The Ghost in the Machine*. Hutchinson, 1967.
17. Pattie Maes. How to do the right thing. Technical Report AIM-1180, MIT Artificial Intelligence Laboratory, October 6 1989.
18. Marco Mamei and Franco Zambonelli. Self-organization in multi agent systems: A middleware approach. In Giovanna Di Marzo Serugendo, Anthony Karageorgos, and et al. Omer F. Rana, editors, *Engineering Self-Organising Systems: Nature-Inspired Approaches to Software Engineering*, number 2977 in LNCS, 2004.
19. Sebastian Rodriguez, Vincent Hilaire, and Abderrafiaa Koukam. Towards a methodological framework for holonic multi-agent systems. In *Proceedings of the ESAW'03 workshop*, pages 179–185, 2003.
20. G Di Marzo Serugendo, N Foukia, S Hassas, A Karageorgos, S Kouadri Mostfaoui, O F Rana, M Ulieru, P Valckenaers, and C Van Aart. Self-organisation: Paradigms and applications. In *Proceedings of ESOA'03*, 2003.
21. S. Forrest Steven and A. Hofmeyr. Engineering an immune system. *Graft*, 4(5):5–9, 2001.
22. Junichi Suzuki and Yoshikazu Yamamoto. Building an artificial immune network for decentralized policy negotiation in a communication endsystem: Openwebserver/inexus study. In *Proceedings of The 4th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2000)*, 2000.
23. Junichi Suzuki and Yoshikazu Yamamoto. A decentralized policy coordination facility in openwebserver. In *¡proceedings of SPA2000*, march 2000.
24. Y. Watanabe, A. Ishiguro, and Y. Uchkawa. Decentralized behavior arbitration mechanism for autonomous mobile robot using immune system. *Books Artificial Immune Systems and Their Applications, Springer-Verlag, p. 186-208, ISBN 3-540-64390-7*, 1999.
25. M. Wooldridge and N. R. Jennings. Intelligent agents: Theory and practice. *The Knowledge Engineering Review*, 10(2):115–152, 1995.
26. Y.Watanabe, A.Ishiguro, Y.Shirai, and Y.Uchikawa. Emergent construction of behavior arbitration mechanism based on the immune system. *Advanced Robotics*, 12(3):227–242, 1998.