How to control emergence of behaviours in a holarchy

Massimo Cossentino
SeT
UTBM
Belfort, FRANCE
Email:massimo.cossentino@utbm.fr
ICAR-CNR, Palermo, ITALY

Stéphane Galland
SeT
UTBM
Belfort, FRANCE
Email: stephane.galland@utbm.fr

Nicolas Gaud
SeT
UTBM
Belfort, FRANCE
Email: nicolas.gaud@utbm.fr

Vincent Hilaire
SeT
UTBM
Belfort, FRANCE
Email: vincent.hilaire@utbm.fr

Abderrafiâa Koukam
SeT
UTBM
Belfort, FRANCE
Email:abder.koukam@utbm.fr

***Abstract*—An open issue in self-organisation is how to control the emergence of behaviour. This issue is also of interest for engineering holonic multi-agent systems as any level of a holarchy is dependant of the emergent behaviours of its sub-levels. In order to tackle this specific feature of holonic multi-agent systems, the capacity concept which abstracts a know-how from its concrete realisation is introduced. The use of this concept is illustrated in this paper through a case study using the ASPECS development process which enables the analysis, design, implementation and deployment of holonic multi-agent systems and integrates the capacity as a core concept of its underlying metamodel.**

I. INTRODUCTION

Works on self-organisation and emergence are common in the Multi-Agent Systems (MAS) field [1]. As pointed out in [2] a still open issue is how to engineer self-organising applications or in other words how to define a global goal, and to design local behaviours so that a global behaviour, able to satisfy this goal, emerges. In [3] the capacity concept is proposed as an abstraction that allows agents and holons to exhibit self-organisation features. This abstraction is integrated in a complete software development process from requirement to code, namely ASPECS. In this paper the way how such an abstraction may be useful to engineer self-adaptation and self-organisation within a Holonic MAS (HMAS) dedicated to the simulation of robot soccer games is shown.

ASPECS is based on a holonic organisational metamodel and provides a step-by-step guide from requirements to code allowing the modelling of a system at different levels of details using a suite of refinement methods. Using a holonic perspective, the designer can model a system with entities of different granularities. He can recursively model sub-components of a bigger system until he achieves a stage where the requested tasks are manageable by atomic easy-to-implement entities. In multiagent systems, the vision of holons is someway closer to the one that MAS researchers have of *Recursive* or *Composed* agents. A holon constitutes a way to gather local and global, individual and collective points of view. A holon is a self-similar structure composed of holons as sub-structures and a hierarchical structure composed of holons is called a *holarchy*. A holon can be seen, depending on the level of observation, either as an autonomous whole entity or as an organisation of holons (this is often called the *Janus effect*, in reference to the two *faces of a holon* [4]).

This paper is organised as follow: section II discusses the methodological background on our holonic framework, the concept of capacity and the ASPECS development process. Section III presents the robot soccer simulation model used to illustrate the emergence control in a holarchy. Related works are discussed in section IV. Finally some conclusions are drawn in section V.

II. METHODOLOGICAL BACKGROUND

A. Holonic Modelling framework

A holon is a whole-part construct that may be composed of other holons, but it may be, at the same time, a component of higher level holons. Examples of holarchies can be found in every-day life. Probably the most widely used example is the human body. The body may be considered as a whole or as a nested hierarchy of organs composed of cells, in turn composed of molecules, etc. Holonic Systems have been already applied to a wide range of applications and a number of models and framework have been proposed for these systems [5], [6], [7]. However, most of them are strongly attached to their domain of application and use specific agent architectures. In order to allow modular and reusable modelling that minimises the impact on the underlying architecture a framework based on an organisational approach is proposed. The Role-Interaction-Organisation (RIO) model [8] is selected to represent organisations since it enables formal specification, animations and proofs based on the OZS formalism [9].

Published in the International Workshop on Self-Adaptation for Robustness and Cooperation in Holonic Multi-Agent Systems (SARC-2008) at the Second International Conference on Self-Adaptive and Self-Organizaing Systems (SASO 2008). October 2008. Venice, Italy..

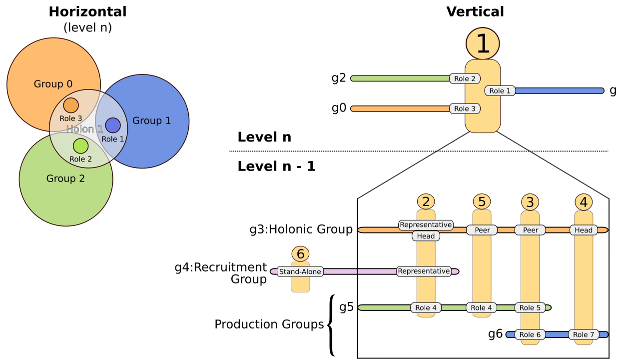

Fig. 1. Horizontal and Vertical decomposition of a holon

In order to maintain this framework generic, two aspects that overlap in a holon should be distinguished. The first is directly related to the holonic nature of the entity (a holon, called super-holon, is composed of other holons, called sub-holons or members) and deals with the government and the administration of a super-holon. This aspect is common to every holon and thus called the *holonic* aspect. It describes the decision making process and how members organise and manage the super-holon. The second aspect is related to the problem to solve and the work to be done. It depends on the application domain and is called the *production* aspect. It describes action coordination mechanisms and interactions between members to achieve the objectives of the super-holons, the tasks to fulfil, or to take a decision.

To manage the first aspect of a composed holon, a particular organisation called *Holonic Organisation* is defined to describe the government of a holon and its structure in terms of authority, power repartition. In [10] three different structures are proposed. The federation where all members of the holon are equals with respect to the interactions with the outside of the holon. The fusion where members disappear to form a single entity and the *moderated group* where a subset of the members are chosen to act as mediator with the outised of the holon. In this work the moderated group is adopted as management structure of the super-holon, due to the wide range of configurations it allows. The *Holonic Organisation* represents a *moderated group* in terms of roles (called *holonic roles*) and their interactions. Three *holonic roles* have been defined to describe the status of a member inside a super-holon and one role to describe the status of non-members: (i) *Representative*, interface of the holon: it is the externally visible part of a super-holon, it is an interface between the outside world (same level or upper level) and the other holon members. It may represent other members in taking decisions or accomplishing tasks (i.e. recruiting members, translating information, …). The *Representative* role can be played by more than one member at the same time. (ii) *Head*, decision maker: it represents a privileged status conferring a certain level of authority in taking decisions inside the holon. This role is not to be confused with the *Representative* role. Indeed, *Representative* role-players are not allowed to take decisions for the holon. Nevertheless, a holon can be at the same time a *Representative* and a *Head*. (iii) *Peer*, default member: Normally in charge of doing tasks assigned by *Heads*, a *Peer* can also have an administrative duty, and it may be employed in the decision making process. It depends on the configuration chosen for modelling the super-holon. (iv) *Stand-Alone*, non-member: This role represents a particular status inside a holonic system. In contrast to the previous holonic roles, it represents the way a member sees a non-member, non composed, holon. *Stand-Alone* holons may interact with the *Representatives* to request their admission as new members of an existing super-holon. Admissions of new holons are problem dependant and can be delt with many different mechanisms.

The three first holonic roles describe the status of a member within a super-holon and participate in defining the holonic organisation. Each of these roles can be played by one or more members, knowing that any super-holon must have at least one *Representative* and one *Head*. The roles *Head*, *Peer* are exclusive between them, while *Representative* may be played simultaneously with one of the two others. Each of these member holonic roles is parameterized using a specific status that specifies if the corresponding holon member is shared between various super-holons. The *Part* status represents members belonging to only one super-holon while the *Multi-Part* status represents sub-holons belonging to more than one super-holon.

In our approach, every super-holon must contain exactly one instance of the *Holonic Organisation*: the holonic group. Every sub-holon must play at least one role in this group to define its status inside the super-holon. Figure 1 illustrates these different aspects of a composed holon and describes its typical structure from an horizontal and a vertical point of view. The vertical decomposition represents the decomposition of a holon, the holon numbered 1 in the figure, in several sub-holons, here 2, 3, 4 and 5. These sub-holons constitutes the level just below the level of holon 1. The horizontal decomposition represents the fact that an holon, whatever the level it belongs to, can play different roles within groups. For example, the holon 2 plays the roles Representative and Head in the holonic group of the holon 1 and other roles in groups 4 and 5.

Super-holons are created with an objective and to perform certain tasks. To achieve these goals/tasks, the members must interact and coordinate their actions. Our framework also offers means to model this second aspect of the super-holons. This goal-dependent interactions are modelled using organisations, namely *Production Organisations* since they are specific to each holon and its goals/tasks. The behaviours and interactions of the members can thus be described independently of their roles as a component of the super-holon. Any number of *production organisations* is possible. Each organisation describes an aspect of the problem dependant aspects of the problem tackled by the holons. The only strictly required organisation is the *Holonic organisation* that describes member's status in the super-holon.

Published in the International Workshop on Self-Adaptation for Robustness and Cooperation in Holonic Multi-Agent Systems (SARC-2008) at the Second International Conference on Self-Adaptive and Self-Organizaing Systems (SASO 2008). October 2008. Venice, Italy..

This approach guarantees a clear separation between the management of the super-holon and the goal-specific behaviours and favours modularity and re-usability.

B. The concept of Capacity

A Capacity describes what a behaviour is able to do or what a behaviour may require to be defined. As a consequence, there are two main ways of using this concept: (i) It specifies the result of some role interactions, and consequently it specifies results that an organisation as a whole may achieve with its behaviour. In this sense, it is possible to say that an organisation may exhibit a capacity. (ii) Capacities may be used to decompose complex role behaviours by abstracting and externalising (for instance by delegating to other roles) a part of their role tasks into capacities. In this case the capacity may be considered as a behavioural building block that increases modularity and reusability of roles and organisations. Thanks to this dual aspect, the capacity thus allows to make the interface between two adjacent levels of abstraction in the organisational hierarchy of the system. A role at level n requires a capacity that is in turn provides by an organisation at level $n-1$.

To better illustrate this dual aspect of the capacity concept, let's consider the example of the capacity to find the shortest path in a weighted directed acyclic graph $\mathcal{G}(N,E)$, from a s source node to a d destination node. This capacity is formally described in figure 3. The template gives the name of the capacity, the required inputs and what is produced as outputs. Constraints on the properties of inputs and outputs are defined by requires and ensures slot respectively. A textual description gives an informal description of the capacity.

Name : FindShortestPath

Input :

- $\mathcal{G} = (N, E)$, directed graph. $E = N \times N$
- $w : E \rightarrow \mathbb{R}$, weight function.
- $s \in N$, source node.
- $d \in N$, destination node.

Output :

$P = \langle s = i_0, i_1, \cdots, i_{n-1}, d = i_n \rangle$, with $\forall k \in \{0..n\}, i_k \in N$

the shortest path P between s and d.

Requires :

$N \neq \varnothing$ and $E \neq \varnothing$ and $\forall (u,v) \in E / w(u,v) \geq 0$

Ensures :

$\forall j_t \in N,\ t \in \{0..m\}$

$$\nexists\, Q = \langle s = j_0, j_1 \cdots, j_m = d \rangle /$$

$$P \neq Q \wedge \sum_{t=0}^{m-1} w(j_t, j_{t+1}) < \sum_{k=0}^{n-1} w(i_k, i_{k+1})$$

There exists no path Q in the graph linking s to d shorter than P.

Textual Description :

provides a solution to the single-source shortest path problem for a directed graph with non-negative edge weights.

Fig. 3. Formal description of FindShortestPath capacity

This capacity may be realised in various ways. *Dijkstra* [11] or *Bellman-Ford* [12] algorithms may be used if the know-how of a single entity is considered. Besides other realisations may be found, especially if the know-how of a group of entities modelled by an organisation is considered. The *Ant Colony* is a well-known organisation able to find a solution to the problem of finding the shortest path in a graph [13]. The solution (the shortest path) emerges from interactions among Ants in their environment. Let us suppose that the environment represents the $\mathcal{G}$ graph, the s source node is mapped to the Ant Hill and the d destination to a food source. Figure 2 represents the design of a portion of a system composed of several levels of abstractions. At the level $n+1$, the *Route Choice* organisation is responsible for providing the best route between two given points to another organisation not represented in the diagram (for instance the *Motion Control* organisation responsible to control the movement of a robot). The request of finding the route is done by the *Route Requester* role (possibly played by a member of the *Motion Control* organisation) responsible to obtain the required information. The route is chosen by the *Route Provider* role that is, indeed, not able to do that by itself, this latter requires the *FindShortestPath* capacity that actually provides the result. This capacity provide the solution of a problem that is effectively solved at a lower level of abstraction (level n). Figure 2 proposes one possible implementation to this capacity by means of an ant colony thanks to the *Ant Colony* organisation that is located at a lower level in the organisational hierarchy. The capacity concept thus allows to define how an organisation at level n may contribute to the behaviour of a role at level $n+1$. Let us consider the need of modelling a complex system behaviour. It is assumed that it is possible to decompose the system from a functional point of view into a set of finer grained (less complex) behaviours interacting to meet the objectives of the organisation. Depending on the considered level of abstraction, an organisation can be seen either as a single behaviour or as a set of interacting behaviours. The concept of organisation is inherently a recursive one [14]. The same duality is also present in the concept of holon. Both are often illustrated by the same analogy: the composition of the human body. The human body, from a certain point of view, can be seen as a single entity with an identity and its own behaviour and personal emotions. Besides, it may also be regarded as a cluster/aggregate of organs, which are themselves made up of cells, and so on. At each level of this nested hierarchy, specific behaviours emerge [15]. The body has an identity and a behaviour that is unique for each individual. Each organ has a specific mission: filtration for kidneys, extraction of oxygen for the lungs or blood circulation for the heart, etc. An organisation is either an aggregation of interacting behaviours, and a behaviour composing an organisation at an upper level of abstraction; the resulting whole constitutes a hierarchy of behaviours that has specific goals to be met at each level.

This recursive definition of the organisation will form the

Published in the International Workshop on Self-Adaptation for Robustness and Cooperation in Holonic Multi-Agent Systems (SARC-2008) at the Second International Conference on Self-Adaptive and Self-Organizaing Systems (SASO 2008). October 2008. Venice, Italy..
2008 © SARC-08

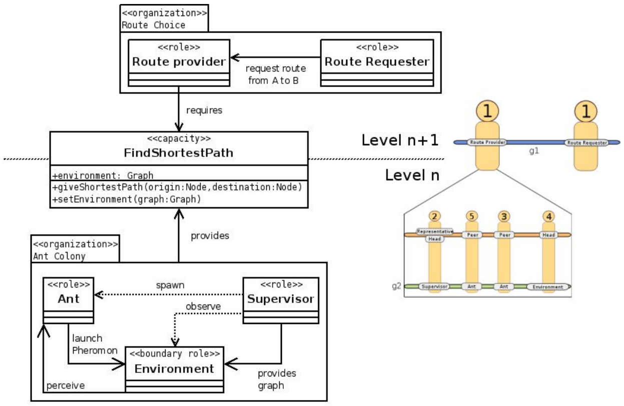

Fig. 2. The concept of capacity as a link between two adjacent levels of abstraction during the analysis

basis of the analysis activities performed within ASPECS development process. The system global behaviour is recursively decomposed into a set of interacting sub-behaviours, each of these latter being in turn decomposed until the lowest level of elementary sub-behaviours is reached. It means that at a given level, composed behaviours are modelled by using organisations. These organisations are composed of roles which can be in turn decomposed as organisations of a lowest level. In most systems, it is somewhat arbitrary as to where the partitioning is left off and what subsystems are taken as elementary (cf. [16, chap. 8]). This choice remains a pure design choice. During the design phase, the hierarchical organisation structure resulting from the analysis will be mapped to a holarchy (hierarchy of holons) in charge of its execution. Each of the previously identified organisations is instantiated in forms of groups. Corresponding roles are then associated to holons to obtain a holarchy able to execute the various behaviours identified during the analysis.

C. The ASPECS development process

The ASPECS software development process can be considered as an evolution of the PASSI [17] methodology for modelling HMAS and it also collects experiences about holon design coming from the RIO approach [8]. The construction of the new process has been performed according to the situational method engineering paradigm [18], [19] and the approach described in [20]. The complete description of the method adopted for building the ASPECS process is out of the scope of this paper. It is sufficient to say that the definition of the MAS metamodel adopted by the new process has been the first step and from this element all the others (activities, guidelines, workflow) have been deducted [20].

The ASPECS process structure (in terms of process metamodel) is based on the Software Process Engineering Metamodel Specification (SPEM) [21] proposed by the OMG. At the core of SPEM is the idea that a software development process is a collaboration between abstract active entities, called *Roles*, that perform operations, called *Activities*, on concrete, tangible entities, called *Work Products*. SPEM clearly separates reusable Method Content from its application in Processes. More precisely a process model is built out of *Process Elements*. Each *Process Element* can be specialised to describe one aspect of a software engineering process. According to this metamodel, the software process of ASPECS is based on three main levels: *Phases*, *Activities* and *Tasks*. A Phase delivers a composite work product (composed of one or more documents that can belong to different work product types), a phase is composed of a number of activities that are in turn decomposable into tasks. An Activity delivers a main work product (like a diagram or a text document) and it is composed of a number of Tasks. A task contributes to the production of a work product (usually by delivering a part of it), and it instantiates/relates/refines MAS metamodel elements.

The ASPECS process is composed of three phases: System Requirements, Agency Society and Implementation and Deployment. The system requirements aims at defining requirements and identifying organisations that will fulfil them. Agency society aims at defining roles, communications and holons architecture. Eventually, implementation and deployment consists in implementing and deploying the concepts of the precedent phase with a dedicated platform. The System Requirement phase is detailed in figure 4. This figure presents the SPEM diagram defining the activities of that phase. Roughly, this phase starts with the depiction of requirements, then the definition of problem ontology and the identification of organisations fulfilling the requirements and the set of roles that compose them. The last activity, named Capacity Identification aims at defining capacities and role dependencies.

III. CASE STUDY

The FIRA Robot soccer competitions began in 1996 using real robots and simulators [22]. It is an example where real-time coordination is needed. Indeed, the principle consists in two teams of five autonomous robots (for the specific case of Mirosot medium league) that play a game similar to human football. It constitutes a well-known benchmark for several research fields, such as MAS, image processing and control. A simulator for such games based upon HMAS using the ASPECS

Published in the International Workshop on Self-Adaptation for Robustness and Cooperation in Holonic Multi-Agent Systems (SARC-2008) at the Second International Conference on Self-Adaptive and Self-Organizaing Systems (SASO 2008). October 2008. Venice, Italy..

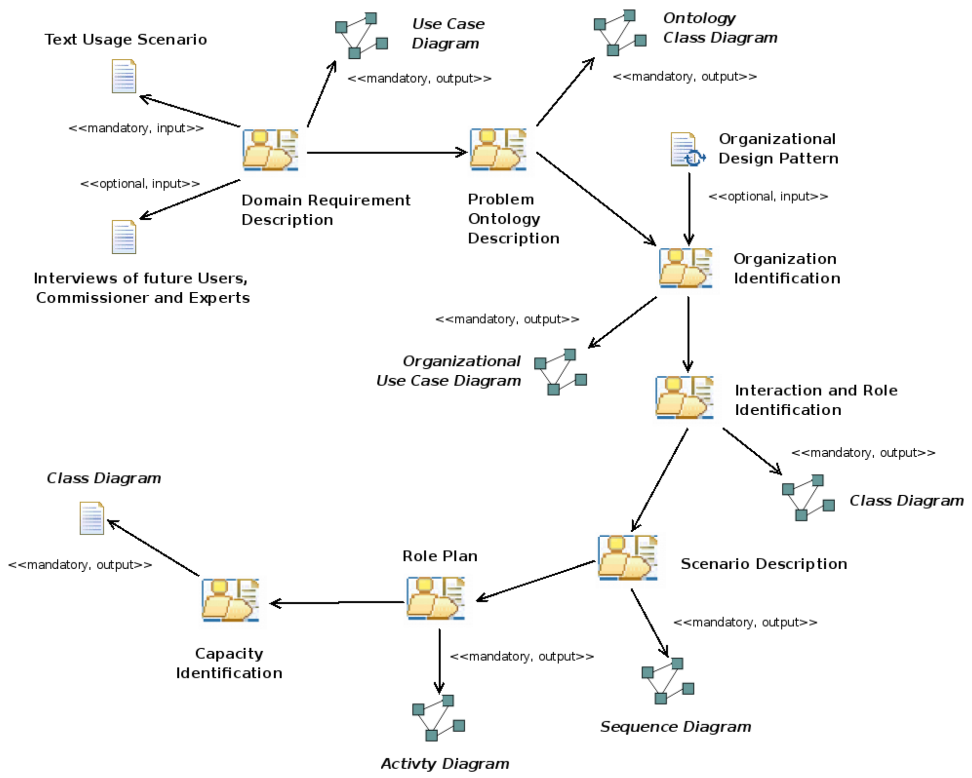

Fig. 4. ASPECS System Requirements Phase

development process was developed. In this paper some parts of the System Requirements phase concerning this simulator are presented. All activities are not detailed since it is out of the scope of this paper. The described activities are Domain Requirement Description, Interaction and Role Identification and Capacity Identification. The interested reader can find the entire case study on the ASPECS website[1].

A. Analysis

The global objective of the Domain Requirements Description (DRD) activity is gathering needs and expectations of application stakeholders and providing a complete description of the behaviour of the application to be developed in terms of functional and non-functional requirements described using an UML use case diagram. Figure 5 details the use cases associated to the development of a simulator for the FIRA Robot Soccer cup. Eight use cases and one actor have been identified. The actor represents the user of the simulator who can simulate matches and tune the strategy of each team. Simulating a match implies the simulation of two autonomous teams that can choose their own strategy and are responsible for simulating the individual robots behaviour. The goal of the Organisation Identification activity is to bind each requirement to a global behaviour, embodied by an organisation. Starting from the results of the DRD activity, use cases are clustered and a first set of organizations that will compose the application is identified following a combination between a structural (or ontological) approach mainly based on the analysis of the problem structure described in the Problem Ontology Description and a functional approach based on requirement clustering. This step relies on analysis decision. In the example described a functional criterion is used and organisations are clustered according to use dependencies, specifically includes relationships. For example, the top level organisation consists in game simulation. This organisation is dedicated to the realisation of simulate matches use case and is composed of simulate team organisation which is dedicated to simulate team's global behavior use case.

According to this first hierarchy of organisations, the objective is now to decompose each organisation in terms of roles and interactions, and precise the behavioural contributions of sub-level organisations to an upper-level ones. In this example a top-down behavioural decomposition is used and for each level a test is done to determine if it is still necessary to continue the decomposition process. Figure 6 describes the result of this decomposition: the various organisations and their respective contributions. At the top of the hierarchy, the *Game Simulation* organisation is decomposed using only one interaction and one role: *Team*. An OCL constraint is added to specify that only two instances of this role are allowed in each instance of this organisation. This role is in charge of simulating the behaviour of a robot soccer team. Its complexity at this level is considered as too high and it will be decomposed into smaller interacting behaviours.

At the second level, the organisation *Team Simulation* is

---

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

Published in the International Workshop on Self-Adaptation for Robustness and Cooperation in Holonic Multi-Agent Systems (SARC-2008) at the Second International Conference on Self-Adaptive and Self-Organizaing Systems (SASO 2008). October 2008. Venice, Italy..

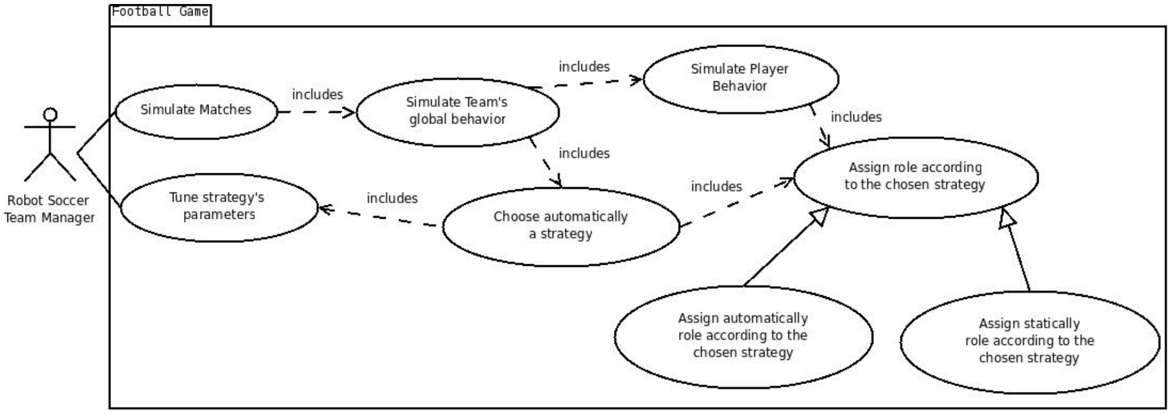

Fig. 5. Domain Requirements Description of the Robot Soccer Simulator Analysis

decomposed in four roles: three roles representing the contributions of previously identified sub-level organisations and a boundary role useful to provide required information about the game situation (players and ball position, score, etc). The *Strategy Selector* role is in charge of determining the best strategy to adopt according to the current situation. Each strategy corresponds to a distribution of strategy roles like *goalkeeper*, *near-defender*, *midfielder*, *shooter*, etc. among the different players. Each robot soccer team is composed of five players. Applying a strategy thus consists in assigning to each player one of the roles defined by the strategy. The association between players and roles defined by the strategy is done by the *Role Assigner* role. The *Players Simulator* role is in charge of simulating the behaviour of the various players according to the chosen strategy. The global state of the game (players and ball positions, score, time, etc.) is maintained by the boundary role *Game Observer*. This role is also in charge of providing required perceptions to the others.

The result of the Interaction and Role Identification activity is thus a three levels holarchy composed of the organisations depicted in figure 6. The analysis is obviously not finished and many decisions have to be taken before being able to implement and deploy the system. The behaviours of the roles need to be defined. The *Strategy Selector* role, for example, is in charge of choosing a strategy for a team according to a game situation. This role is of an uttermost importance for the team efficiency and the choice of a strategy is not so trivial. However there exist several techniques in order to exhibit such a behaviour.

An arbitrary choice is made to abstract the choice of a strategy with a capacity namely *ChooseStrategy* capacity. This strategy takes as input a game situation and a non empty set of strategies and returns a single strategy as result. At this step, this capacity may be considered as a new requirement to identify sub-level organisations able to fulfil it. The next subsection presents an organisation based upon the theory of immune system which enables to implement this capacity.

B. *The Immune System Organisation*

From a computational viewpoint the human immune system can be viewed as a parallel, distributed system that has the capacity to control a complex system over time [23]. The human immune system is composed of several layers of defence such as: physical (skin), innate and adaptive. The adaptive part of the human immune system has been mostly taken into consideration in this paper. The adaptive system improves its response to a specific pathogen at each exposure. Thus, the adaptive system has three key functionalities: recognition, adaptation and memory. The adaptive immune system can be divided into two major sections: the humoral immune system and the cellular immune system. The former acts against antigens by means of proteins called immunoglobulins or antibodies which bind to antigen. This binding mechanism allows an antibody to either tag an antigen for attack by other part of the immune system or neutralise antigens. The latter, among other duties, destroys virus-infected cells.

Among numerous theories which try to explain the human immune system, Nobel Laureate N. K. Jerne proposed a model based on interactions between antibodies [24]. These interactions take the form of stimulation and inhibition. This theory is known as Jerne's Idiotypic Network. The network is defined by stimulation/inhibition links between antibodies. From now on, immune system term will be used as a reference to the immune network. The region by which antibodies stimulate or inhibit other antibodies is called idiotope. Idiotopes play the roles of antigens for other antibodies. It means that each antibody may be seen by other antibodies as an antigen if its idiotope corresponds to the paratope of these antibodies. This regulation mechanism enables the immune system to maintain an effective set of cells and to self-organise in order to deal with antigens. Indeed, the stimulation/inhibition links are based on affinities between antibodies to deal with specific antigens. If two types of antibodies are able to match two similar type of antigens then they will have affinity and will stimulate each other. On the contrary if antibodies are built to deal with very different types of antigens they will inhibit themselves.

Published in the International Workshop on Self-Adaptation for Robustness and Cooperation in Holonic Multi-Agent Systems (SARC-2008) at the Second International Conference on Self-Adaptive and Self-Organizaing Systems (SASO 2008). October 2008. Venice, Italy..

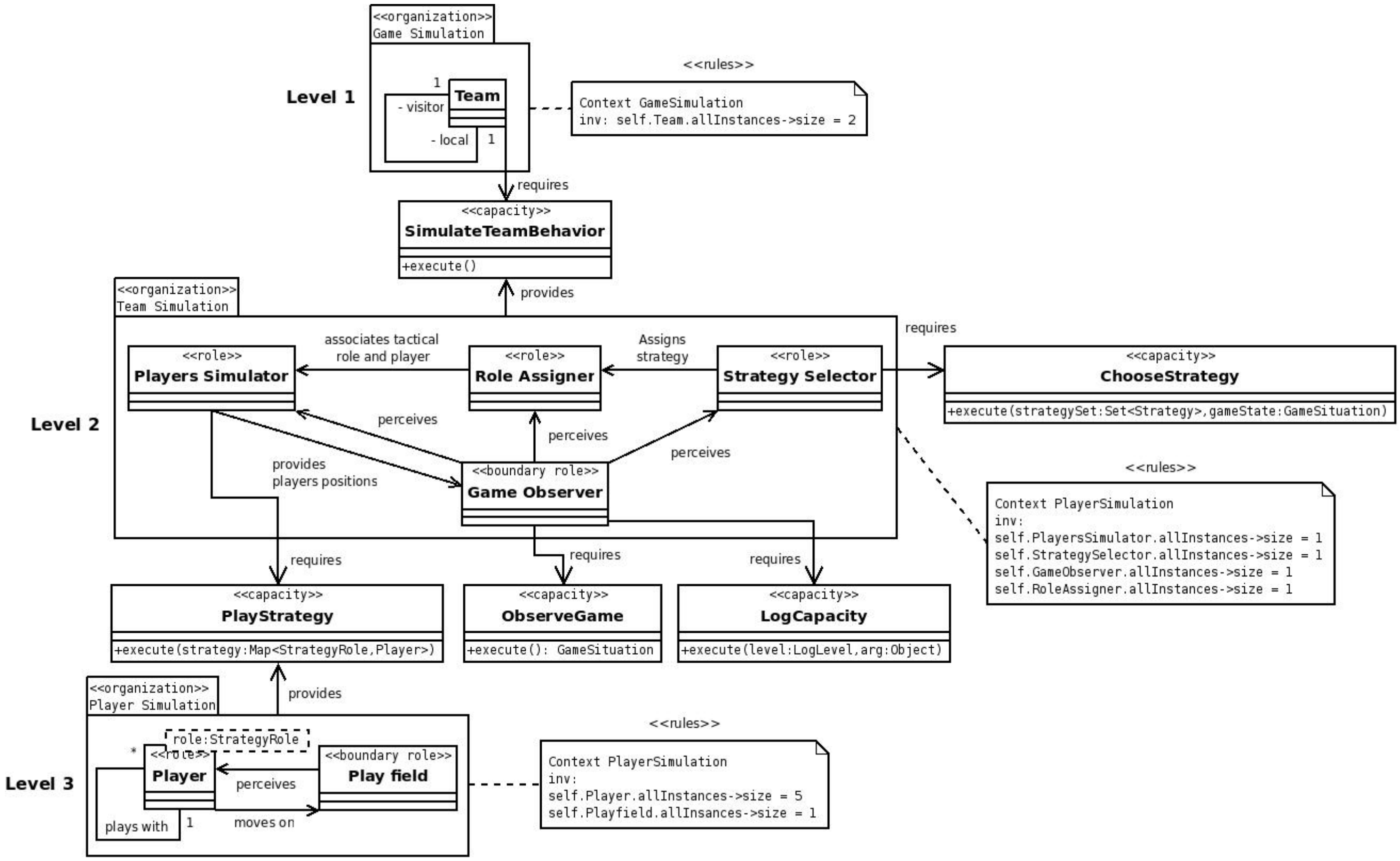

Fig. 6. Results of Interaction and Role Identification, and Capacity Identification activities of the Robot Soccer Simulator Analysis

Jerne's Idiotypic Network has already been used as agent architecture, for example in [25]. Such architecture is an interpretation of the Jerne's theory. Concepts developed in that work are used for a single agent case as a basis for the approach presented in this paper. The main principle of this architecture is that each antibody represents a possible behaviour of the agent with its preconditions and affinities with other antibodies. It is an arbitration mechanism which allows both the choice of a single behaviour according to some stimulations, the antigens, and learning with the continuous computation of affinities between these antibodies. In our case the behaviour will be associated to a strategy and the choice of a behaviour is then equivalent to the choice of a strategy.

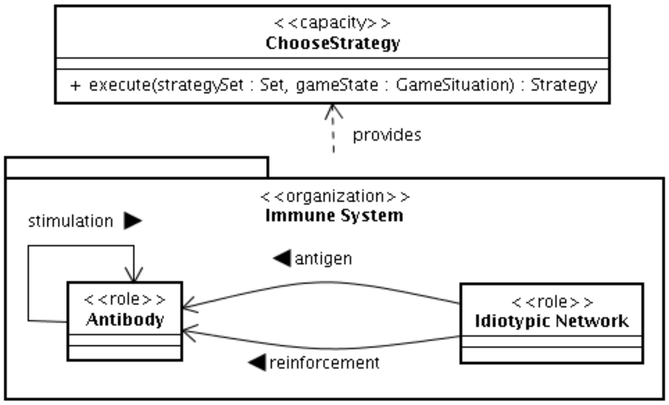

Fig. 7. Organisational model of the Artificial Immune System

Figure 7 describes the organisation model of the immune system. It is composed of two roles: *Idiotypic Network* and *Antibody*. The Antibody role describes the behaviour exhibited by *Antibodies* in the network. *Antibodies* influence each other through the *affinities interaction*. When appropriate, *antibodies* will send stimulation/inhibition stimuli, according to the affinities values of the *antibody*, to other *antibodies*. With these stimulations/inhibitions and the present antigens, each *antibody* computes its concentration. The *antibodies* then send their concentrations to the *Idiotypic Network* for selection. The *antibody* with the greater concentration is chosen and executes its behaviour. After execution the results are analysed by the *Idiotypic Network* which sends rewards or penalties to antibodies in order to update their affinities. Figure 8 details the interactions resulting from the perception of an antigen by the immune system. The perception and encoding of antigens is done by an analysis of the environment. The perceived antigens are sent to all antibodies. Each antibody checks if it is stimulated by the antigen and if stimulated it broadcasts its affinity values. It means that a stimulated antibody will

Published in the International Workshop on Self-Adaptation for Robustness and Cooperation in Holonic Multi-Agent Systems (SARC-2008) at the Second International Conference on Self-Adaptive and Self-Organizaing Systems (SASO 2008). October 2008. Venice, Italy..

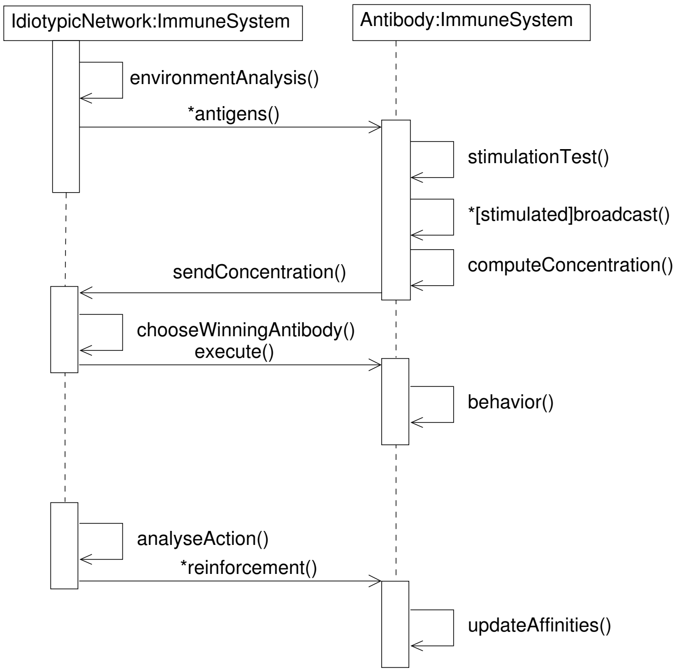

Fig. 8. A complete step of the Immune System interactions

Fig. 9. Minimum distance between robots and ball

either stimulates or inhibits other antibodies.

C. Self-Adaptation

The major reason for the choice of the idiotypic network as a realisation of the *ChooseStrategy* capacity is that idiotypic networks exhibit self-adaptation characteristics. In the case of complex systems such as robot soccer games this kind of technique is very helpful as it is very difficult to decide a priori, without knowledge about the opponent team, heuristics for strategy choice. As reported in [26] experiments conducted with the example described in this paper has proven that the architecture is able to self-adapt to game situations. Figure 9 presents some results from these experiments. It shows that the team using idiotypic networks develops the ability to control the ball. Here it is measured with the minimum distance between robots and the ball. Of course other techniques could have been used in place of idiotypic networks such as learning architectures, BDI-agent, etc. In fact, it is one objective of the capacity concept to abstract from the concrete realisation of a know-how, for choosing a strategy. The capacity ChooseStrategy can then be realised by say a set of cooperating BDI-agents. It can be even changed at runtime without hindering the system functionalities.

Moreover, in our case the realisation of the capacity is the result of roles interaction within an instance of the *Immune System* organisation. It means that the realisation of the capacity is done by a group of holons. The capacity is thus a mean to control the emergence of behaviour. The *Immune System* organisation can be considered as an organisational design pattern which solves the problem of choosing a specific element of a set with reinforcement learning mechanism associated to this choice.

IV. RELATED WORKS

Several approaches related to agent capabilities have been already proposed in various domains of MAS.

In the domain of Semantic Web and Web Agents, [27], [28] propose an Agent Capability Description Language (LARKS) and discuss the Service Matchmaking process using it. Thus a first description of Agent Capability using LARKS is given. However this description is only used in the Service Matchmaking process and not used during the analysis nor the modelling phases. These aspects are tackled in our approach with the notion of capacity as a basic description of an agent know-how.

In [29] the concept of capability and the concept of opportunity, respectively representing the necessary and the sufficient conditions to achieve a goal. These concepts are similar to the concepts of capacity and capacity realisation but they do not take into account organisational nor holonic aspects.

To distinguish the agent from its competencies, [30] and [31] have introduced the notion of *skill* to describe basic agent abilities and allowing the definition of an atomic agent, that can dynamically evolve by learning/acquiring new *skills*. Then [32], [33] have extended this approach to integrate this notion of *skill* as a basic building block for role specification. [34], [35] also consider agent capability as a basic building block for role specification in their meta-model for MAS modelling. These capabilities are however inherent to particular agents, and thus to specific architectures. In these models the role is considered as a link between agents and a collection of behaviours embodied by the skills. In other words, in these approaches, the skill is directly related to the way to obtain a service, and thus represents a basic software component. However, the description of the general class of related services and the fact that a given agent ability can be obtained by various implementations is not developed. This really differs from our view of the notion of role. For us a role is a first class entity, the abstraction of a behaviour or/and a status in an organisation (extension of [8]), that should be specified without making any assumptions on its susceptible players. It is assumed that these aspects are captured in our model with the notion of capacity implementation.

In a more general way, our approach is situated in the confluence of these various models, linking the description of an agent capability and its various possible implementations. Agents are thus provided with means to reason about their needs/goals and to identify the way to satisfy/achieve them. It benefits us the advantages of both approaches, increasing reusability and modularity by separating the agent from its capacities, and the capacity from its various implementations.

Considering an organisation as a possible capacity implementation constitutes one of our main contributions; group of interacting agent can provide a capacity to an upper level.

Published in the International Workshop on Self-Adaptation for Robustness and Cooperation in Holonic Multi-Agent Systems (SARC-2008) at the Second International Conference on Self-Adaptive and Self-Organizaing Systems (SASO 2008). October 2008. Venice, Italy..

This takes all its interest in the case of holonic MAS, where the super-holon can exploit additional behaviours emerging from members interactions to obtain a new capacity. In the same way, a modelling tool to deal with intrinsic emergent properties of a system and to catch them directly from the analysis phase is provided.

V. CONCLUSION

In this paper a case study illustrating the use of the capacity concept and how this concept enables to control the emergence of behaviour is presented. The chosen case study consists in a simulator for robot soccer games and it is considered under a holonic perspective. The capacity concept is part of the metamodel underlying the ASPECS development process. The entire set of activities of this process is out of the scope of this paper so only relevant activities applied to the case study are presented. This paper emphasises the intrinsic ability of ASPECS to catch the various levels of abstraction of a complex system; this occurs during the analysis phase by using an organisational hierarchy and in the design phase by using an holarchy.

REFERENCES

[1] G. D. M. Serugendo, N. Foukia, S. Hassas, A. Karageorgos, S. K. Mostefaoui, O. F. Rana, M, P. Valckenaers, and C. V. Aart, "Self-organisation: Paradigms and applications," in *Proceedings of ESOA'03*, 2003.

[2] G. D. M. Serugendo, M.-P. Gleizes, and A. Karageorgos, "Self-organisation and emergence in mas: An overview," *Informatica*, vol. 30, no. 1, pp. 45–54, 2006.

[3] S. Rodriguez, N. Gaud, V. Hilaire, S. Galland, and A. Koukam, "An analysis and design concept for self-organization in holonic multi-agent systems," in *Engineering Self-Organising Systems* (S. Bruckner, S. Hassas, M. Jelasity, and D. Yamins, eds.), vol. 4335 of *LNAI*, pp. 15–27, Springer-Verlag, 2007.

[4] A. Koestler, *The Ghost in the Machine*. Hutchinson, 1967.

[5] F. Maturana, *MetaMorph: an adaptive multi-agent architecture for advanced manufacturing systems*. PhD thesis, The University of Calgary, 1997.

[6] M. Ulieru and A. Geras, "Emergent holarchies for e-health applications: a case in glaucoma diagnosis," in *IEEE IECON 02*, vol. 4, pp. 2957–2961, 2002.

[7] J. Wyns, *Reference architecture for Holonic Manufacturing Systems - the key to support evolution and reconfiguration*. PhD thesis, Katholieke Universiteit Leuven, 1999.

[8] V. Hilaire, A. Koukam, P. Gruer, and J.-P. Müller, "Formal specification and prototyping of multi-agent systems," in *ESAW* (A. Omicini, R. Tolksdorf, and F. Zambonelli, eds.), no. 1972 in LNAI, Springer Verlag, 2000.

[9] P. Gruer, V. Hilaire, A. Koukam, and P. Rovarini, "Heterogeneous formal specification based on object-z and statecharts: semantics and verification.," *Journal of Systems and Software*, vol. 70, no. 1-2, pp. 95–105, 2004.

[10] C. Gerber, J. Siekmann, and G. Vierke, "Holonic multi-agent systems," Tech. Rep. DFKI-RR-99-03, DFKI - GmbH, 1999.

[11] E. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, pp. 269–271, December 1959.

[12] R. Bellman, "On a routing problem," *Quarterly of Applied Mathematics*, vol. 16, no. 1, pp. 87–90, 1958.

[13] N. Attiratanasunthron and J. Fakcharoenphol, "A running time analysis of an ant colony optimization algorithm for shortest paths in directed acyclic graphs," *Inf. Process. Lett.*, vol. 105, no. 3, pp. 88–92, 2008.

[14] J. Ferber, *Multi-Agent Systems. An Introduction to Distributed Artificial Intelligence*. London: Addison Wesley, 1999.

[15] G. Chauvet, *La vie dans la matire*. Flammarion, 1998.

[16] H. A. Simon, *The Science of Artificial*. Cambridge, Massachusetts: MIT Press, 3rd ed., 1996.

[17] M. Cossentino, "From Requirements to Code with the PASSI Methodology," in *Agent-Oriented Methodologies* (B. Henderson-Sellers and P. Giorgini, eds.), ch. IV, pp. 79–106, Hershey, PA, USA: Idea Group Publishing, 2005.

[18] B. Henderson-Sellers, "Method engineering for OO systems development," *Commun. ACM*, vol. 46, no. 10, pp. 73–78, 2003.

[19] C. Rolland and N. Prakash, "A proposal for context-specific method engineering," in *Proc. of the IFIP TC8, Working conference on method engineering on Method engineering : principles of method construction and tool support*, pp. 191–208, Chapman & Hall, Ltd., 1996.

[20] M. Cossentino, S. Gaglio, A. Garro, and V. Seidita, "Method fragments for agent design methodologies: from standardization to research," *International Journal on Agent Oriented Software Engineering*, vol. 1, pp. 91–121, April 2007.

[21] SPEM, *Software Process Engineering Metamodel Specification, v2.0, Final Adopted Specification, ptc/07-03-03*. Object Management Group, March 2007.

[22] Y. H. Kim, *Micro-robot world cup soccer tournament*. KAIST, 1996.

[23] J. D. Farmer, N. H. Packard, and A. S. Perelson, "The immune system, adaption and machine learning," *Physica D*, vol. 22, pp. 187–204, 1986.

[24] N. K. Jerne, "Towards a network theory of the immune system," *Ann Immunol (Inst Pasteur)*, vol. 125C, pp. 373–389, 1974.

[25] 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.

[26] V. Hilaire, A. Koukam, and S. Rodriguez, "An adaptative agent architecture for holonic multi-agent systems," *ACM Transactions on Autonomous and Adaptive Systems*, vol. 3, no. 1, pp. 1–24, 2008.

[27] K. Sycara, M. Klusch, S. Widoff, and J. Lu, "Dynamic service matchmaking among agents in open information environments," *SIGMOD Record (ACM Special Interests Group on Management of Data)*, vol. 28, pp. 47–53, March 1999.

[28] K. Sycara, J. Lu, M. Klusch, and S. Widoff, "Matchmaking among heterogeneous agents on the internet," in *Proceedings of the 1999 AAAI Spring Symposium on Intelligent Agents in Cyberspace*, March 1999.

[29] L. Penserini, A. Perini, A. Susi, and J. Mylopoulos, "From capability specifications to code for multi-agent software," in *ASE*, pp. 253–256, IEEE Computer Society, 2006.

[30] J. Routier, P. Mathieu, and Y. Secq, "Dynamic skill learning: A support to agent evolution," in *AISB'01*, pp. 25–32, 2001.

[31] M. Savall, M. Itmi, and J.-P. Pecuchet, "Yamam : a new organization model for multi-agent systems and its platform named phoenix," in *Conference SCSC 2000*, (Orlando, USA), 2001.

[32] E. Adam and R. Mandiau, "A hierarchical and by role multi-agent organization: Application to the information retrieval.," in *ISSADS*, pp. 291–300, 2005.

[33] E. Adam and R. Mandiau, "Roles and hierarchy in multi-agent organizations," in *CEEMAS 2005* (M. Pechoucek, P. Petta, and L. Varga, eds.), no. 3690 in LNAI, (Budapest, Hungary), pp. 539–542, Springer-Verlag, September 2005.

[34] E. Matson and S. A. DeLoach, "Autonomous organization-based adaptive information systems," in *KIMAS '05*, (Waltham, MA), April 2005.

[35] E. Matson and S. A. DeLoach, "Formal transition in agent organizations," in *IEEE International Conference on Knowledge Intensive Multiagent Systems (KIMAS '05)*, (Waltham, MA), April 2005.

Published in the International Workshop on Self-Adaptation for Robustness and Cooperation in Holonic Multi-Agent Systems (SARC-2008) at the Second International Conference on Self-Adaptive and Self-Organizaing Systems (SASO 2008). October 2008. Venice, Italy..