

An Empirical Approach for Relating Environmental Patterns with Agent Team Compositions

Mariana Ramos Franco^{1(✉)}, Gustavo A.L. Campos²,
and Jaime Simão Sichman¹

¹ Laboratório de Técnicas Inteligentes, Escola Politécnica,
Universidade de São Paulo, São Paulo, Brazil
ramos.franco@gmail.com, jaime.sichman@poli.usp.br

² Mestrado Acadêmico em Ciência da Computação,
Universidade Estadual do Ceará, Fortaleza, Brazil
gustavo@larces.uece.br

Abstract. The design of a rational organization composed of a team of agents is a challenging problem in domains such as collective robotics, cyber warfare, war games and military missions. In these domains, the team is designed to confront an opponent team with technical and numerical equivalence, and aiming to conquer areas where there are scarce resources of high economic value, that are distributed in locations within a territory whose topology is unknown. In these scenarios, it is hard for the agents to do the right thing. In addition to being competitive, the task environment is unknown, partially observable and dynamic. The challenge is how to design a rational team whose members are not ideal rational agents. This work argues that one approach is to implement a suitable organizational specification that fits the task environment, according to some previously defined environmental patterns that include both domain-specific and topological characteristics. In this work, we present an experimental evaluation of these patterns' influence on the performance of teams of agents evolving on the Agents on Mars scenario, a well-known agent programming testbed. The results of the evaluation show that organizational specifications that exploit this information perform better than others that don't.

Keywords: Organizations · Team formation · Engineering Multi-Agent systems

1 Introduction

The design of rational agents is a nontrivial task, especially in hard task environments. The hardest case corresponds to a multi-agent, unknown, partially observable, non-deterministic, sequential, continuous and dynamic environment [1]. The designer must be satisfied with a non-ideal rational agent to evolve in these domains. In a multi-agent domain, the challenge that arises is how to design the organization of the multi-agent system; more specifically, in competitive environments, the problem to solve is how to design a rational team of agents, each of them not ideally rational [2, 3].

In our context, a MAS organization is "... a supra-agent pattern of emergent or predefined cooperation of the agents in the system, that could be defined by the designer or by the agents themselves, to achieve a purpose." [2]. The notion of team identifies an organizational paradigm where a group of agents must work together to achieve a common goal in a task environment. Each agent assumes a role and commits himself to attain some goals that are necessary to achieve the team's overall objective. The team maintains an explicit representation of its organization, called Organizational Specification (OS) [3].

A team of agents must be composed of some specialist agents to be able to operate in hard environments, such as exploring unknown places as foreign planets in a type of war where the ground squads realize campaigns for the installation of forces in resource-rich areas. The number of individual agents of each capability in a heterogeneous team is its composition. In the organizational context, each agent assumes a role in the team and the team of agents can be divided into two or more groups of agents. The team adequate structure and composition, i.e., the number of groups in the team and the specialist agents in each group, are not known a priori in these hard environments, and will produce poor performance for the team if they were not done properly.

This work is a contribution for the sub problem of designing the teams' organizational structure, what is called in organization theory as the "synthesis problem", namely: "which structures are best suited to solve optimally certain types of problems?" [4], or "given a certain set of conditions to be satisfied, how to find the network which is best?" [5]. The problem considers that the designer knows in which kind of environment the team should evolve and his task is to determine a suitable organizational structure that fits the task environment.

We consider that a suitable team's organizational structure must fit the task environment in each domain. We focus on the relation between some environmental patterns in the task environment and the number of groups/squads of agents in the team structure, specially to assess the influence of this relation in the performance of a team designed to confront an opponent team with technical and numerical equivalence, and aiming to conquer areas where there are scarce resources of high economic value (clusters), that are distributed in locations within a territory whose topology is unknown.

We consider that the same approach can be applied in other domains in which the environment can be represented by a weighted graph, where the vertexes denote resources and possible locations for the agents, and the edges indicate the possibility of crossing from one vertex to another with a cost for the agent. In our work, a cluster is a "valuable area" represented by any connected subgraph in which the resource values in the vertices are greater than an inferior limit value of resource by vertex, and the sum of all values in the vertices is greater than an inferior limit value of resource by cluster. The notion of environmental pattern is defined based on some clusters' spatial attributes, whose values can be perceived by the teams when they explore the environment.

The main contribution of this paper is a methodological one – viz. an empirical approach for relating features of task environments to successful agent structural organization. In the context of the team composition, we consider that the designer has worked previously on the team's OS; his task is to refine the initial structural

specification, determining which roles the agents can play in the groups, and how many groups and their cardinality are necessary to maximize the team's performance evaluation measure in an environment with previously known environment patterns.

The agents' team performance evaluation is domain-dependent. In our case, it is based on the *Agents on Mars* scenario, a testbed provided by the Multi-Agent Programming Contest [6]. The results of our evaluation show that organizational specifications that exploit this information perform better than others that don't. For example, the number of clusters leads, in some cases, to situations where it may be better for the whole team to occupy a single cluster, while in other cases it may be better to divide the team into smaller squads to try to gain control over multiple clusters in the environment.

2 Evaluation of the Organizational Design

Previous research in the field of organization theory has already focused on the evaluation of the quantitative effect of the organizational design on the SMA's performance.

Horling and Lesser [7] focused on the comparison of organizational paradigms. They present a survey of the major organizational paradigms used in multi-agent systems. These include teams and others human organizational patterns. They provided the descriptions of these patterns, their advantages and disadvantages, and examples of use. They argue that their work allows the designers to recognize a large set of structural possibilities, to realize comparative evaluation of organizational structures and then to select an appropriate organizational design for a domain and environment.

van der Broekwork et al. [8] proposed an approach for the analysis and the formal modeling of agent-based organizations. The approach addresses both the organization structure and its dynamics. The environment is considered as a special component of the organization model. It serves as a source of events for the organization. The environment is populated by agents, which under certain conditions may be allocated to organizational roles. By performing simulations and verification, the approach provides formal techniques and tools for different types of analysis of organization models.

Hodgson et al. [9] developed a framework that supports hierarchical modeling of teams of agents. A team (group) is a composite component that is characterized by several roles, which are enacted by agents and other teams. The framework focuses on the technical side of programming and implementation of SMA. The authors introduced a formal language for specifying the dynamics of individual roles and teams, providing different interesting types of analysis of the SMA dynamics.

Scerri et al. [10] have focused on studies about the properties and the performance of teams with hundreds of members. They developed a model of teamwork to address the limitations of others models applied to very large teams. The model organizes team members into sub teams that evolve dynamically, and into overlapping sub teams that work on sub goals of the overall team goal. They experimented these very large teams evolving in distinct domains, such as control of fire trucks responding to an urban disaster and simulated unmanned aerial vehicles in a battle space.

Grossi et al. [4, 11] argue that "organizational structures should be seen along at least three dimensions: power, coordination, and control". They provided a technical

terminology for describing the notion of structural organization and its properties. The concepts are defined rigorously by means of concepts from graph theory. In addition to be useful for describing the organizational structure, they can be employed to provide a formal analysis of the effect of such structures on the activities in the SMA. Their formal tool can be useful to provide numerical analyses of the organizational structures, and for evaluating to what extent an organizational structure exhibits some characteristic properties such as robustness, flexibility and efficiency.

Machado et al. [12] developed a detailed discussion related to multi-agent patrolling and an empirical evaluation of possible patrolling solutions in domains such as computer networks and computer war games. The authors proposed different architectures of multi-agent systems, various evaluation criteria, applied in two experimental settings. They implemented a patrol simulator. The results show that some kinds of architecture can patrol an area, in certain circumstances, more adequately than others.

Furtado and Filho [13] described a simulator of crimes in an urban area. The user configures and allocates police forces in certain geographical regions and then interacts with the simulation, watching the crime behavior in the presence of preventive police. They described how studies involving simulations can help to determine whether a reorganization process is necessary and how it should be performed. The simulation results illustrate how to exploit opportunities for the system to be reorganized.

Although some of these detailed studies about structures had tried to answer the question of which structures are best suited to solve problems in environments that can change (or not) frequently, they do not address other properties that are intrinsic in hard task environments, as the case of a partially observable and unknown environment, neither any kind of analysis relating to patterns detected in this environment.

3 The Suitable Team Structure

The design of a rational agent team can be a very complex problem. The hardest case corresponds to an environment that is partially observable, nondeterministic, sequential, continuous, unknown and dynamic [1]. One of the main challenges associated with the design problem, involves the search for solutions for the synthesis problem, namely, “which structures are best suited to solve optimally certain types of problems?” [4]. In this context, the relation between the set of possible patterns of the environment and the set of possible organizational structures for a team is a necessary information for the designer of a rational team, i.e., that must be able to use this knowledge to revise the team formation, changing for a suitable structure when the team discovers the environment patterns or when these patterns change [14].

3.1 Environmental Patterns

We focus in this work on agent teams designed to confront opponent teams with technical and numerical equivalence, and aiming to conquer areas where there are scarce resources of high economic value, that are distributed in locations within a territory whose topology is unknown, but that can be discovered by the teams as the agents interact with their environment. The agent teams’ territory can be a physical

environment, as a battlefield in an unconventional war, where military groups are formed in order to realize campaigns to conquer certain areas in big cities, which were contained by traffickers and militias [15]. It can be a virtual battlefield in a war electronic game, multiplayer, online first-person shooter [16]. The battle can happen in the cyberspace as well, where usually anonymous cybernetic attacks occur, that are directed at political leaders, military systems, and any ordinary citizen, anywhere in the world [17].

These battlefields can be represented by a vertex-edge-labeled graph (weighted graph with two vertex functions). Formally, this vertex-edge-labeled graph $G = (V, E)$ consists of a set of vertices V and a set of edges E , which contains unordered pairs of distinct elements of V . Each vertex in the graph has a resource value and, optionally, an identifier. The resource value associated with a vertex and its identification are modeled respectively by a vertex value function $vvf: G(V) \rightarrow \mathbb{N}$ and by a vertex labeling function $vlf: G(V) \rightarrow \text{Idv}$, where \mathbb{N} is the set of natural numbers employed to value the resources in the vertices in V , and Idv is a subset of the natural numbers (e.g.: $\{1, \dots, |V|\}$) employed to identify the $|V|$ vertices in V . Each edge in the graph indicates the possibility of crossing from one vertex to another. The cost of edge crossing for the agent is modeled by an edge value function $evf: G(E) \rightarrow \mathbb{R}$, where \mathbb{R} is the set of real numbers employed to value the costs in the edges in E .

The notion of environmental patterns in this work tries to capture the diversity of environments, i.e., how many “valuable areas”, subgraphs representing the higher valued resources, appear in a graph G representing a battlefield. In our work, these “valuable areas” are called clusters. We consider that a cluster is any connected sub-graph C of G in which the resource values in the vertices are greater than an inferior limit value of resource by vertex, and the sum of all values in the vertices is greater than an inferior limit value of resource by cluster.

To formally define the notion of cluster, let infVc denotes the inferior limit value of resource capacity by vertex in a cluster, and infCc the inferior limit value of resources capacity by cluster. We define the notion of cluster in the graph $G = (V, E)$ as any sub-graph $C = (V', E')$, such $V' \subseteq V$ and $E' \subseteq E$, that satisfies three conditions:

- (1) $\forall v \in C(V'), vvf(v) \geq \text{infVc}$
- (2) C is a connected sub-graph
- (3) $\text{infCc} \leq \sum_{v \in C(V')} vvf(v) \leq \sum_{v \in G(V)} vvf(v)$

The first condition ensures that the resource value associated with each vertex in a cluster C is greater than the inferior limit value of resource capacity by vertex. The second condition ensures that the clusters in the graph G can be a sub-graph C with only one (isolated) vertex with high value, or a sub-graph in which all vertices in C are not isolated. Finally, the third condition ensures that the sum of the resource values associated with all vertices in a cluster C is greater than the inferior limit value of resources capacity by cluster, that must be lesser than or equal the graph capacity, i.e., the sum of all values of resource capacity in the graph G .

Figure 1 illustrates the notion of a cluster $C = (V', E')$ in a graph $G = (V, E)$, when $|V| = 21$ vertices and $|E| = 27$ edges. The vertex identification was omitted in the figure. The vertices that are not graphically described by a circle in Fig. 1(a) represent

the vertices v in V that satisfy the first condition to be a member of V' when $\text{InfVc} = 1$, i.e., $\text{vvf}(v) \geq 1$. The three selected sub-graphs in Fig. 1(b) represent those sub-graphs G' in G that satisfies the first and the second condition to be a cluster C in G . The two selected sub-graphs in Fig. 1(c), C1 and C2, represent the sub-graphs G' in G that are clusters, i.e., those that satisfy the three conditions when $\text{InfCc} = 10$, i.e., $\sum_{v \in C1(V1)} \text{vvf}(v) \geq 10$ and $\sum_{v \in C2(V2)} \text{vvf}(v) \geq 10$. When considering the distribution of values of vertices of a cluster C in G , two extreme cases are possible: highly diverse clusters with the maximum possible number of distinct values, and homogeneous clusters having vertices of the same value.

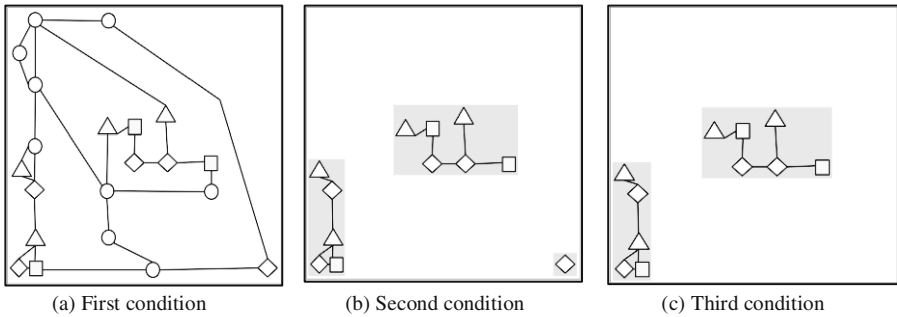


Fig. 1. Clusters

The notion of environmental pattern associated with a vertex-edge-labeled graph G is defined as a tuple $\text{EP}(G) = \langle S, N, H, D \rangle$, considering four attributes associated with the distribution of all clusters in G , such that: S (size) represents the mean number of vertices of the clusters in G ; N (number) represents the quantity of clusters in G ; H (homogeneity) is a boolean value which indicates if clusters in G have the same value; and D (dispersion) represents the approximate mean distance in number of vertices between the clusters in G . So, we have at least eight extreme case associated with these four attributes. The vertices in the clusters in Fig. 1(c) have different resource values, so that the value of: a circle is one, a square is two, a triangle value is three and a lozenge is four. Therefore, the environmental pattern associated with the graph in the figure is: $S = 5, N = 2, H = 0$ e $D = 7$. Figure 2 shows graphs with different cluster patterns, setting the parameters in the cluster conditions as $\text{infVc} = 1$ and $\text{infCc} = 10$.

The larger vertices in black color identify the vertices and edges that satisfy the three cluster's conditions. Figure 2(a)–(b) illustrate different sizes for the clusters. Figure 2(c)–(d) illustrates different number of clusters, three clusters in SC3 and five clusters in SC4. The graphs in Fig. 2(e)–(f) have different homogeneity values, i.e., the vertices in the clusters in SC5 have approximately the same value and the vertices in SC6 have not. Finally, different dispersion values are illustrated in Fig. 2(g)–(h), i.e., where the six clusters in SC7 are more dispersed on the graph and the six clusters in SC8 are more close to each other.

Our work considers that the attainment of a goal by a team involves a relation between its goal, its organizational specification and its external environment. The

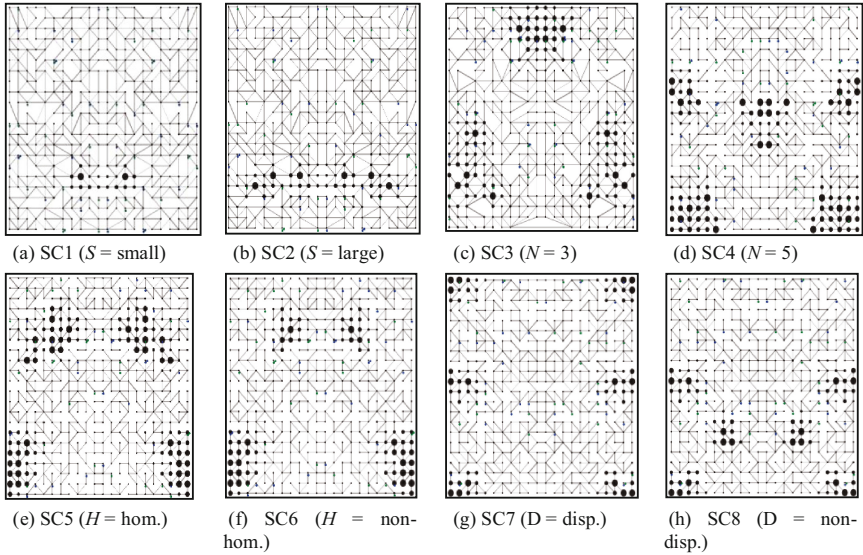


Fig. 2. Environments with different patterns

success of a team depends on their internal construction and on where it is placed to function. If the team's organizational specification is suitable for its external environment, or vice versa, the team will reach the desired objective. Moreover, from the point of view of the structural specification, in a great number of cases, the attainment of the team's goal depends on a few features in the external environment, and is independent of its details. We focus on the relation between the environmental patterns in the task environment and the number of groups/squads of agents in the team structure, especially to assess the influence of this relation in the performance of a team designed to confront an opponent team with technical and numerical equivalence.

3.2 Teams' Structural Organization

A team is a human organizational pattern that can be characterized by a set of cooperative agents, which have agreed to work together attempting to maximize their utility [18]. The designer must solve two related problems to design an agent team: the design of individual cooperative heterogeneous agents and the design of a rational team organization for these agents. The design of each kind of agent requires the formalization of the function that maps each sequence of perceptions into an action, that is necessary to realize tasks like to find and to conquer the best clusters, to defend these clusters, to attack the agents in the adversary team and to help the agents in the same team.

In relation to the agent organization, the designer must formally describe the team's organizational specification OS. In this work, this notion is based on the MOISE organizational modeling language [19, 20]. It decomposes the organizational specification $OS = \langle SS, FS, DS \rangle$ in three dimensions. The Structural Specification (SS) defines the

roles, relationships between roles, and the team’s groups. The Functional Specification (FS) defines the global goals and how these are decomposed into subgoals and missions. The deontic specification (DS) relates these two dimensions, identifying subsets of missions and goals in FS that are permitted and/or required for each role in SS.

The MOISE Structural Specification SS is built in three levels. Let Agents represents the set of N agents in the team, and Roles the set of M roles that can be played by any agent $A_i \in \text{Agents}$. Let $\text{AgR}_j = \{A_i \mid A_i \in \text{Agents plays a role } R_j \in \text{Roles}\}$. At the individual level, the SS defines the set of behaviors that an agent $A_i \in \text{AgR}_j$ is responsible for when he adopts the role R_j . At the social level, the SS defines three kinds of links between roles: authority, communication and acquaintance. An authority link $\text{link}(A_i, A'_i, \text{aut})$ implies the existence of a communication link $\text{link}(A_i, A'_i, \text{com})$, the latter implies the existence of an acquaintance link $\text{link}(A_i, A'_i, \text{acq})$. For example, in the case where the link type is acquaintance, an agent source $A_i \in \text{AgR}_j$, playing a source role R_j , has a representation of the agent destination $A'_i \in \text{AgR}'_j$, playing a destination role R'_j . At the collective level, the SS defines the aggregations of roles in groups.

A group is created from a group specification indicating the subset of roles that should be played in the group and their respectively min-max cardinality (how many agents can play those roles), its set of subgroups and their respectively min-max cardinality and the sets of intra-group and inter-group links [20]. Figure 3(a) illustrates the MOISE structural specification of a team T composed of groups of agents Gr (min = 1; max = 8), that can play two roles C (min = 1; max = 8) and S (min = 0; max = 7), and one group of one agent that can play one role L (min = 1; max = 1).

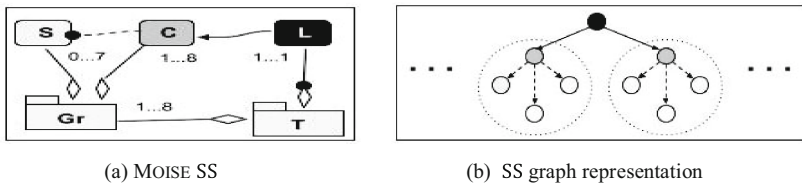


Fig. 3. An agent team’s organizational structure

Hence, a MOISE structural specification SS can be viewed as a vertex-labeled graph $G = (V, E)$, whose vertexes V are playing role agents and whose edges E represent authority, communication and acquaintance links. Figure 3(b) illustrates the previous SS graph representation. The graph in Fig. 3(b) is one of the valid representations of the team organizational structures that can be generated from the MOISE SS in Fig. 3(a), where the structure of each sub-group Gr is represented by a tree, generated by the authority inter-group link between role L (agent in the black vertex) and role C (agents in the grey vertices), and the communication intra-group link between roles C and role S (agents in white vertices). Section 4.2 presents a concrete example to clarify some key points provided in this section in an abstract way.

3.3 The Environmental Patterns and the Teams' Structural Organizations

From the point of view of team composition, a team of agents able to operate in a hard environment, rather than being a homogeneous team, must be composed of specialist agents. In the organizational context, in the organizational context, each agent assumes a role in the team and the team of agents can be divided into two or more groups of agents. The adequate number of individual agents of each capability in each group and the number of groups in a heterogeneous team are not known a priori in the scenarios approached in our work. Discovering the optimal composition and structure are relevant features in the team design as it may avoid poor performance.

Our work is a contribution for the sub problem of designing the team's organizational structure, that is, which structures of specialist agents are best suited to solve optimally certain types of problems. We consider that the designer has worked previously on the team's organizational specification $OS = \langle SS, FS, DS \rangle$, specifying which roles the agents can play in the groups and their links (SS), the team's current goals in its missions, the max-min number of agents that can commit with them (FS), and the agents that can or should commit to the goals and missions (DS). Given these previous information, the team composition task is to find a graph $Go = (Vo, Eo)$ that represents a suitable organizational structure in an environment represented by the graph $Ge = (Ve, Ee)$ with a known pattern $EP(Ge) = \langle S, N, H, D \rangle$.

More specifically, to find Go the designer must solve two related choice problems: (a) to choose the number of groups of specialist agents and (b) to choose the cardinality of each group, such as the team structural specification SS is not violated and aimed to maximize the team's performance evaluation measure in the environment Ge . We believe that the knowledge about the relation between the set of possible patterns for the environment and the set of possible organizational structures for a team can be learned by some agents in the organization to eventually revise the team formation; this may be done when patterns are discovered in a partially observable and unknown environment, or when they change in a dynamic environment. We describe in the sequence how we experiment different structural specifications with distinct environmental patterns possibilities.

4 Experimental Evaluation

Our experimental approach consists of proposing different organizational specifications that are suitable for different environmental patterns (EPs), and can be adapted to generate knowledge about the agent teams' performance.

4.1 Evaluation Scenario

The application domain used in this work is the Agents on Mars scenario, developed in the Multi-Agent Programming Contest [6]. The task environment consists of a battlefield in an unconventional war, in which artificial agents have special sensors and actuators to explore and to conquer rich areas in natural resources, which are in

unknown places. Although it seems a specific type of war application, where the ground squads realize campaigns for the installation of forces in the rich resource areas, the chosen specific scenario has the major components that occur in campaigns in a diversity of application domains.

The scenario environment shown in Fig. 4 is represented by a graph where the vertices denote water wells and possible locations for the agents, and the edges indicate the possibility of crossing from one vertex to another with an energy cost. A zone is a sub-graph covered by a team per a coloring algorithm based on the notion of domain [6]. Agents from different teams can be in a single vertex. The team with the highest number of agents dominates the vertex, which receives the dominant team color. An uncolored vertex inherits the color from its neighborhoods dominant team. If the graph contains a sub-graph with a colored border, all the nodes that are within this boundary receive the same color.

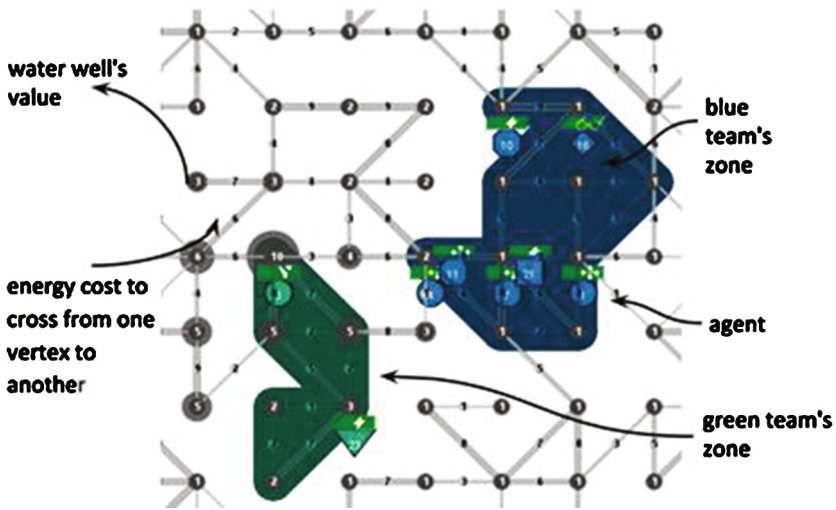


Fig. 4. Agents on Mars Scenario

At the beginning of the simulation, the map is unknown to the agents. Each team consists of 28 players that can be of five different types: explorers, sentinels, saboteurs, inspectors and repairers. These types define the characteristics of each agent such as life level, maximum energy, strength, and visibility. The roles also limit the possible actions that the agent can perform in the environment. For instance, explorers can discover water wells and help to explore the map, while sentinels have long-distance sensors and thus can observe larger areas, saboteurs can attack and disable enemies, inspectors can spy opponents, and repairers can repair damaged agents.

A team receives a cash reward whenever it reaches a major milestone. This reward can be used to empower the agents, increasing their maximum energy or strength. Different milestones can be reached during a competition, such as dominating areas with fixed values, having performed a successful number of attacks or well-succeeded

defenses. If not used, the reward is added to the team’s total score. The goal of each team is to maximize its score, defined as the sum of the values obtained by the occupied zones with the earned (and not yet spent) rewards in each step of the simulation.

4.2 Agent Teams

In our work [27], we decided to design BDI agents and to give emphasis on the functioning of the teams’ organizations in graphs with different EPs. Each agent is composed of plans, a belief base and its own world model, that consists of a graph. It captures every detail received from the environment, such as explored vertices and edges, opponents’ position, disabled teammates, etc. Figure 5 illustrates the block diagram adopted in the design of all agents in our team.

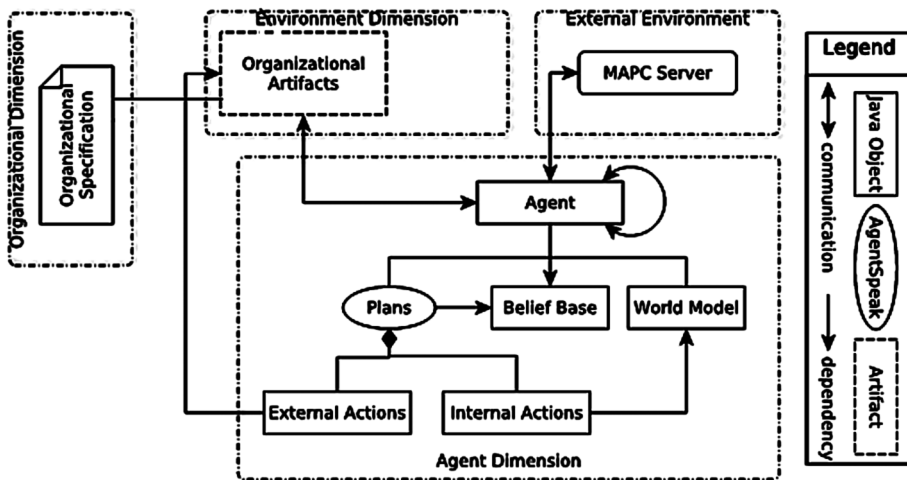


Fig. 5. Agents’ Architecture

The world model consists of a graph representing the scenario environment. It captures every detail received from the environment (MAPC contest server). At each step, the agent’s world model is updated with the percepts received from the MAPC server, and with the information received from the other agents. Some of the percepts received from the MAPC server are also stored in the agent’s belief base, such as the agent’s type, energy, position and team’s rewards, thus allowing the agent to have a direct access to this information without access its world model. The agent decides which plan will be executed considering its beliefs and the local view of the world. In addition to the agents’ types defined by the Scenario, we defined additional different roles in our system. Table 1 describes the mission related to each role.

Each of these roles has a mission associated with it, and can be played by one or more agents. The coordinator is a kind of agent internal to our system, which does not communicate with the MAPC server. Whenever the world model is updated, he computes which are the best zones in the graph and send this information to the other

Table 1. Types and missions of agents

Type	Mission
Explorer	Explores the whole graph by probing every vertex and surveying all edges on its path
Soldier	Tries to occupy one of the best zones indicated by the coordinator agent
Guardian	Defends the squad by attacking any opponent that is close to the team’s zone
Medic	Repairs the agents in the squad
Soldier exp.	Explores the team’s zone by probing the vertices whose values are unknown
Saboteur	Attacks any close opponent
Sentinel	Sabotages the opponent by moving inside its zone
Reparer	Repairs the saboteur and the sentinel
Coordinator	Builds its local view of the world through the percepts broadcasted by the other agents

agents. Although he determines the best areas of the map, each agent decides for himself which empty vertex he will occupy to form a zone or increase it. The coordinator is also responsible for creating the organizational specification, in the beginning of a competition, and for distributing the groups, roles and missions among the other agents. Figure 6 shows a diagram to illustrate one possible team’s SS.

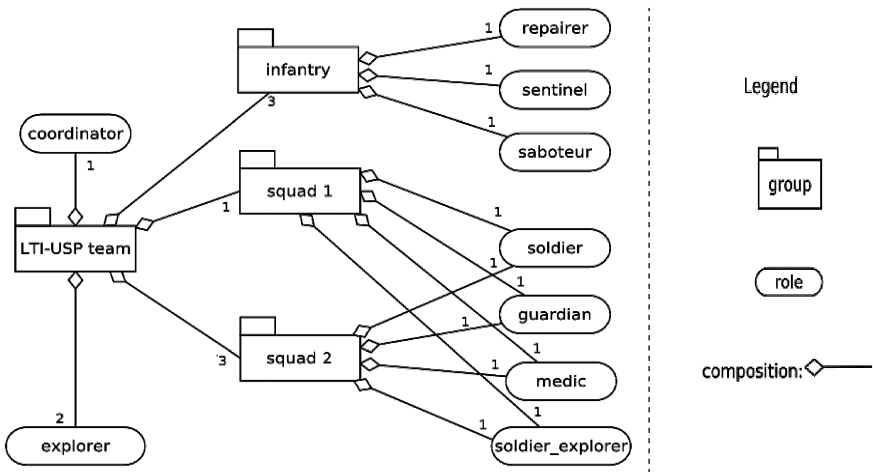


Fig. 6. Team structural specification

This team is composed of three infantry group, one coordinator, two explorers and the other sixteen agents divided into four squads. Each squad had nearly the same number of agents, and respected the number of agents of each type when assigning the roles. The team was developed using a platform for MAS programming called JaCaMo [21], which supports all levels of abstractions – agent, environment, and organization – that are required for developing sophisticated MAS, by combining three separate technologies: Jason for programming autonomous agents [22], CArTAgO for

programming environment artifacts [23, 24], and MOISE for programming multi-agent organizations [25].

Jason is a Java-based interpreter for an extended version of the AgentSpeak programming language, for programming BDI agents. CArTAgO is a framework for environment programming based on the A&A meta-model [24]. In CArTAgO, the environment can be designed as a dynamic set of computational entities called artifacts, organized into workspaces, possibly distributed among various nodes of a network [21]. Each artifact represents a resource or a tool that agents can instantiate, share, use, and perceive at runtime. We made use of the organizational artifacts provided in Moise.

4.3 Description of the Experiments

The goal of our experiments was to evaluate the impact of the environmental patterns over teams' performance, by modifying the environmental pattern $EP(G) = \langle S, N, H, D \rangle$ in the Agent on Mars map, and measuring the performance of two adversary teams composed by the same BDI agents, but with two different organizational specifications $OS = (SS, FS, DS)$. For each team, the performance measure adds the values of its conquered clusters with the earned rewards, as the team reaches a major milestone. The experiments tried to perceive the impact of the EP over teams' performance.

The experiments [26] consisted of seven teams: TG1, TG2, TG3, TG4, TG6, TG8 and TG10. These seven teams competed in 14 environments with different EPs, including the eight scenarios illustrated in Fig. 1. In relation to teams' OS, we fixed the attribute values associated with the organizational dimensions FS and DS, and modified the number of squads and the cardinality of each squad in the structural dimension SS. Each team TGN is composed of one infantry group, two explorers and the other 23 agents divided into n squads. Each squad had nearly the same number of agents, and respected the number of agents of each type when assigning the roles.

Each team played against each other three times, and the team that wins most matches wins the overall tournament. Each match had 750 steps and the map was randomly generated. From one match to another we changed the number of vertices, edges and high-valued areas. For each environment, we performed 10 simulations for each of the possible matches. The data collected in all simulation was the winner and the final score of each team. These metrics were used to indicate the performance of each team. We used the Wilcoxon T test as a hypothesis test to define for each match if the 10 simulations were sufficient or not to conclude that a team was better than other in a determined environment. The results of the Wilcoxon T test for the environment SC3 (Fig. 1) is shown in Table 2.

Table 2. Results of the Wilcoxon T test for SC3

	TG1	TG2	TG3	TG4	TG6	TG8	TG10
TG1		0,0059	0,0195	0,0020	0,0039	0,0371	0,0137
TG2	0,0059		0,6953	0,0488	0,0273	0,7695	0,6250
TG3	0,0195	0,6953		0,1602	0,0371	0,3750	0,0371
TG4	0,0020	0,0488	0,1602		0,1602	0,0840	0,0273
TG6	0,0039	0,0273	0,0371	0,1602		0,1602	0,2324
TG8	0,0371	0,7695	0,3750	0,0840	0,1602		0,6250
TG10	0,0137	0,6250	0,0371	0,0273	0,2324	0,6250	

The Wilcoxon T test is a non-parametric test for dependent samples that can indicate with some stated confidence level if a particular population tends to have larger values than other. Based on the results of the Wilcoxon T test, it's possible to represent the partial order of a certain number of teams given their performance in a specific environment by a partial order graph. Figure 7 illustrates the partial order graph obtained by drawing an edge from team A to team B whenever a significant difference ($p - \text{value} < 0.05$) exists between the performances of two teams.

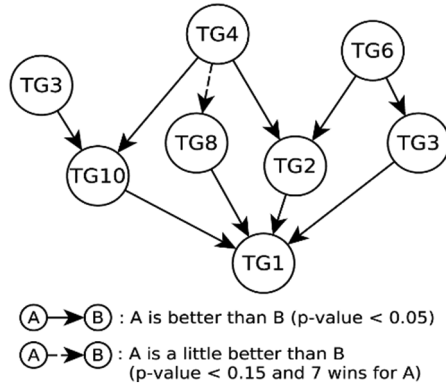


Fig. 7. Example of a partial order graph

The direction of the edge will be from the team with better performance to the team with lower performance, and we omit the edges that can be extrapolated through a transitive closure, that is, those edges for which there is already a path connecting two teams. The dashed edges represent the matches for which the Wilcoxon T test came close to detect a significant difference between two teams ($p - \text{value} < 0.15$ and at least 7 wins for one team). Finally, it's possible that the same team appears twice or more times in graph as shown in Fig. 5, in which TG3 appears twice. This is necessary so that we can represent that TG3 achieved a better performance than TG10, but the same was not observed for TG6 and TG10. This analysis was performed on all environments and the results obtained are presented in the following subsections.

4.4 Obtained Results

Here, a cluster is any subgraph of the maps formed by vertices with value greater than 1 (infVc), in which the sum of all vertices is greater than 10 (infCc). In relation to clusters' size (S), in general, a team with a number of squads equal to the number of zones (a zone is a sub-graph covered by a team according to a coloring algorithm based on the notion of domain [6]) in the cluster does better. In the environment SC2 it's possible to divide the cluster in 6 zones to be occupied by the squads and, as shown in Fig. 8(a), TG6 is better than almost all the other teams, because: (i) a smaller number of squads is not effective in securing and defending bigger clusters, and (ii) a larger number ends up by causing some squads to disperse out of the cluster, making it difficult to conquer it.

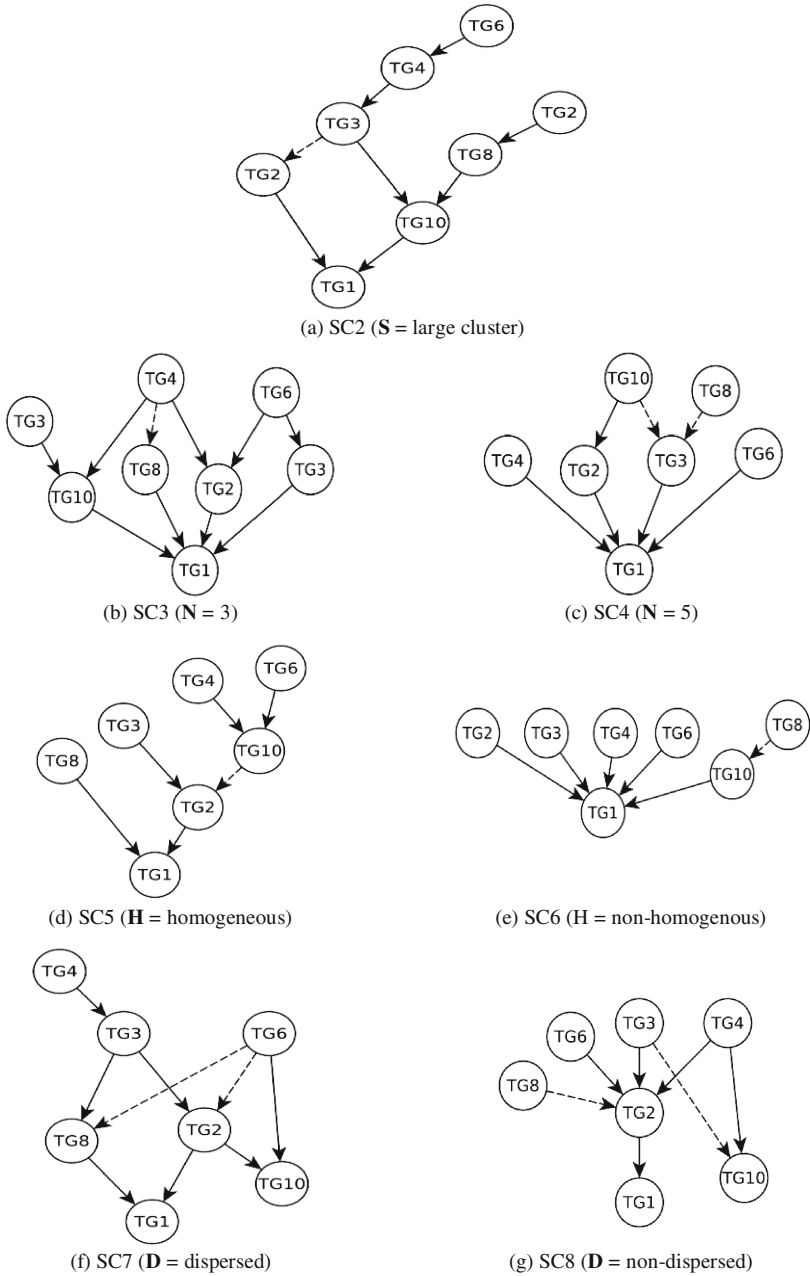


Fig. 8. Partial order graphs for different environments

Regarding the number of clusters (N), the team must have a number of squads equal or closer to the number of clusters on the map. If the number is smaller, the team will not cover all good areas, which can then be easily occupied by the opponent. If the number of squads is greater than the number of clusters, some agents will be placed in areas of small values, weakening the squads' attack and defense. In Fig. 8(b) it is possible to see that for the environment SC3 with 3 clusters, TG3 performs better than TG10, while TG10 outperforms TG3 for SC4, which has 5 clusters.

Regarding the homogeneity (H), the experiments showed that finding and occupying the clusters with the highest values is critical in non-homogeneous environments. In these cases, it is better to balance among the teams (Fig. 8(c)) since the winner ends up being defined by the team that occupies the bigger clusters. This is good for teams with small number of squads, because there is a chance that they occupy the clusters with highest values, while the opponent with a larger number of squads ends up by spreading its agents in smaller and not valuable clusters.

Regarding the clusters' dispersion (D), the results showed that less dispersed clusters help teams with a larger number of squads to form larger areas than they could if the clusters (and the agents) were dispersed (Fig. 8(d)). Moreover, with the proximity of the clusters, the squads tend to be closer of each other, making it easier to attack the opponent. For example, TG3 and TG4 can dominate large areas in SC7, which often comprise two neighbor clusters, while TG8 occupies smaller areas and does not attack efficiently the opponents. However, in SC8, with the proximity of the clusters, TG8's squads tend to be closer, making it easier to attack the opponent, and thus helping TG8 to occupy sometimes areas that cover more than one cluster at the same time.

5 Conclusions and Further Work

Our evaluation tried to assess the impact of environmental patterns in the performance of a team of agents, designed to confront an "equivalent" opponent team, aiming to conquer areas with scarce resources of high economic value, in unknown locations in the environment. In our work, these areas were denoted as clusters and were employed to generate the notion of environmental pattern, which was essential to operationalize the evaluation.

We performed the evaluation considering the impact of the environmental patterns in a valued graph map over the performance of teams with the same functional and deontic specifications, but with a different structural specification, due the different numbers of squads and the cardinality of each squad in the team. Although our results are preliminary ones, we believe that they provide at least two contributions that can be exploited in the design of agents' teams when the task environment is hard, but that can be described in terms of environmental patterns.

The first contribution is related to the knowledge the designer can learn about these task environments, in order to assess whether a team will be able to selectively search for solutions in the map. The results of our evaluation provided some knowledge about the different organizational specifications that are suitable to different environmental patterns in the maps. In this sense, we hope to intensify this initial evaluation in two directions: (1) from the environment' side, to evolve scenarios that aggregate

diversified environmental patterns, obtained from the combination of the different attribute values employed to represent patterns; and (2) from the team's side to extend the approach to other attribute values associated with the three dimensions employed to define organizational specifications.

The second contribution is related to the proper concretization of the notion of environmental pattern realized in the work. Considering this notion as a complementary representation of the state of the environment, and the consistent knowledge that can be provided by a more intensive evaluation of the impact of environmental patterns over the performance of teams, we believe that two possibilities are generated for the designer (or any artificial agent properly designed): (1) to predict the behavior of a team if he knows its goals, its organizational specification, and the current environmental patterns; (2) to define a suitable organizational specification for a team and the properties of its behavior if the agent knows the current environment pattern and team's goal. These are hypotheses that we hope to prove soon.

Acknowledgements. Jaime Simão Sichman is partially supported by CNPq, Brazil, grant agreement no. 303950/2013-7.

References

1. Russel, S., Norvig, P.: *Artificial Intelligence: A Modern Approach*, 3rd edn. Pearson Education, London (2010)
2. Boissier, O., Sichman, J.S.: Organization oriented programming. In: *Tutorial Notes*, 3rd. International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2004), New York (2004)
3. Coutinho, L., Sichman, J., Boissier, O.: Modeling dimensions for agent organizations. In: Dignum, V. (ed.) *Handbook of Research on Multi-Agent Systems: Semantics and Dynamics of Organizational Models*. IGI Global, Hershey (2009)
4. Grossi, D., Dignum, F., Dignum, V., Dastani, M., Royakkers, L.: Structural aspects of the evaluation of agent organizations. In: Noriega, P., Vázquez-Salceda, J., Boella, G., Boissier, O., Dignum, V., Fornara, N., Matson, E. (eds.) *COIN -2006*. LNCS, vol. 4386, pp. 3–18. Springer, Heidelberg (2007). doi:[10.1007/978-3-540-74459-7_1](https://doi.org/10.1007/978-3-540-74459-7_1)
5. Harary, F.: Status and contrastatus. *Sociometry* **22**, 23–43 (1959)
6. Behrens, T., Köster, M., Schlesinger, F., Dix, J., Hübner, J.F.: The multi-agent programming contest 2011: a résumé. In: Dennis, L., Boissier, O., Bordini, R.H. (eds.) *ProMAS 2011*. LNCS, vol. 7217, pp. 155–172. Springer, Heidelberg (2012). doi:[10.1007/978-3-642-31915-0_9](https://doi.org/10.1007/978-3-642-31915-0_9)
7. Horling, B., Lesser, V.: *A Survey of Multi-Agent Organizational Paradigms*. Computer Science Technical Report 04-45. University of Massachusetts (2004)
8. Broek, E.L., Jonker, C.M., Sharpanskykh, A., Treur, J., Yolum, P.: Formal modeling and analysis of organizations. In: Boissier, O., Padget, J., Dignum, V., Lindemann, G., Matson, E., Ossowski, S., Sichman, J.S., Vázquez-Salceda, J. (eds.) *AAMAS 2005*. LNCS, vol. 3913, pp. 18–34. Springer, Heidelberg (2006). doi:[10.1007/11775331_2](https://doi.org/10.1007/11775331_2)
9. Hodgson, A., Roennquist, R., Busetta, P., Howden, N.: Team Oriented Programming with Simple Team. In: *Proceedings of SimTecT 2000*, Sydney, Australia, pp. 115–122 (2000)
10. Scerri, P., Xu, Y., Liao, E., Lai, J., Sycara, K.: Scaling teamwork to very large teams. In: *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2004)*. ACM Press, New York (2004)

11. Grossi, D., Dignum, F., Dastani, M., Royakkers, L.: Foundations of organizational structures in multiagent systems. In: Proceedings Fourth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2005), pp. 690–697. ACM Press, Utrecht (2005)
12. Machado, A., Ramalho, G., Zucker, J.-D., Drogoul, A.: Multi-agent patrolling: an empirical analysis of alternative architectures. In: Simão Sichman, J., Bousquet, F., Davidsson, P. (eds.) MABS 2002. LNCS, vol. 2581, pp. 155–170. Springer, Heidelberg (2003). doi:[10.1007/3-540-36483-8_11](https://doi.org/10.1007/3-540-36483-8_11)
13. Furtado, V., Filho, E.J.: A multi-agent simulator for teaching police allocation. In: Proceedings of the 17th conference on Innovative applications of artificial intelligence (IAAI 2005) - Volume 3, pp. 1521–1528 (2005)
14. Lu, H.: Team formation in agent-based computer games. In: Pisan, Y. (ed.) Proceedings of the 2nd Australasian Conference on Interactive Entertainment, IE 2005, Sydney, Australia, pp. 121–124. ACM Press (2005)
15. Melo, A., Belchior, M., Furtado, V.: Analyzing police patrol routes by simulating the physical reorganization of agents. In: Sichman, J.S., Antunes, L. (eds.) MABS 2005. LNCS, vol. 3891, pp. 99–114. Springer, Heidelberg (2006). doi:[10.1007/11734680_8](https://doi.org/10.1007/11734680_8)
16. The Gamasutra Quantum Leap Awards: First-Person Shooters, GamaSutra. http://www.gamasutra.com/view/feature/1832/the_gamasutra_quantum_leap_awards_.php. Accessed 8 Feb 2016
17. Czosseck, C., Geers, K. (eds.): The Virtual Battlefield: Perspectives on Cyber Warfare. Cryptology and Information Security Series, vol. 3. IOS Press, Amsterdam (2009)
18. Carley, K., Gasser, L.: Computational organization theory. In: Weiss, G. (ed.) Multiagent Systems, pp. 299–330. MIT Press, Cambridge (1999)
19. Hübner, J.F., Sichman, J.S., Boissier, O.: A model for the structural, functional, and deontic specification of organizations in multiagent systems. In: Bittencourt, G., Ramalho, G.L. (eds.) SBIA 2002. LNCS, vol. 2507, pp. 118–128. Springer, Heidelberg (2002). doi:[10.1007/3-540-36127-8_12](https://doi.org/10.1007/3-540-36127-8_12)
20. Hübner, J., Sichman, J., Boissier, O.: Developing organised multiagent systems using the MOISE + model: programming issues at the system and agent levels. *Int. J. Agent-Oriented Softw. Eng.* 1–27 (2007)
21. Boissier, O., Bordini, R.H., Hübner, J., Ricci, A., Santi, A.: Multi-agent oriented programming with JaCaMo. *Sci. Comput. Program.* **78**(6), 747–761 (2011)
22. Bordini, R.H., Hübner, J., Wooldridge, M.: Programming Multi-agent Systems in AgentSpeak using Jason. Wiley-Blackwell, Chichester (2007)
23. Ricci, A., Piunti, M., Viroli, M.: Environment programming in multi-agent systems: an artifact-based perspective. *Auton. Agents Multi-Agent Syst.* **23**(2), 158–192 (2010)
24. Omicini, A., Ricci, A., Viroli, M.: Artifacts in the A&A meta-model for multi-agent systems. *Auton. Agents Multi-Agent Syst.* **17**(3), 432–456 (2008)
25. Hübner, J., Boissier, O., Kitio, R., Ricci, A.: Instrumenting multi-agent organisations with organisational artifacts and agents. *Auton. Agents Multi-Agent Syst.* **20**(3), 369–400 (2009)
26. Franco, M.R.: Avaliação Organizacional de Times de Agentes para o Multi-Agent Programming Contest. Master’s thesis. Escola Politécnica da Universidade de São Paulo (2014)
27. Franco, M.R., Sichman, J.S.: Improving the LTI-USP team: a new JaCaMo based MAS for the MAPC 2013. In: Cossentino, M., Fallah Seghrouchni, A., Winikoff, M. (eds.) EMAS 2013. LNCS, vol. 8245, pp. 339–348. Springer, Heidelberg (2013). doi:[10.1007/978-3-642-45343-4_19](https://doi.org/10.1007/978-3-642-45343-4_19)