International Journal of Intelligent Information Systems
2014; 3(6-1): 15-22
Published online October 17, 2014 (http://www.sciencepublishinggroup.com/j/ijiis)
doi: 10.11648/j.ijiis.s.2014030601.13
ISSN: 2328-7675 (Print); ISSN: 2328-7683 (Online)

A new organization model for self-organizing multi-agent systems based on self-adaptation features

Amin Rahmanzadeh, Ali Farahani, Eslam Nazemi

Self-* Laboratory, Faculty of Electrical and Computer Engineering, Shahid Beheshti University, Tehran, Iran

Email address:
a.rahmanzadeh@mail.sbu.ac.ir (A. Rahmanzadeh), a_farahani@sbu.ac.ir (A. Farahani), nazemi@sbu.ac.ir (E. Nazemi)

To cite this article:
Amin Rahmanzadeh, Ali Farahani, Eslam Nazemi. A New Organization Model for Self-Organizing Multi-Agent Systems Based on Self-Adaptation Features. *International Journal of Intelligent Information Systems.* Special Issue: Research and Practices in Information Systems and Technologies in Developing Countries. Vol. 3, No. 6-1, 2014, pp. 15-22. doi: 10.11648/j.ijiis.s.2014030601.13

Abstract: Complexity of information technology systems is increasing continually. A very good solution would be using agents in implementing and controlling these systems. Multi-Agent Systems are good examples of using agents for system control and implementation. On the other hand, Multi-Agent systems need to be controlled and managed too. Using organizations is one of the best solutions. Many research studies have been done in this field. In this paper we will try to introduce, explain, and compare some related works. We will extract their pros and cons. For the weaknesses of this research studies, we have proposed some solutions. In this paper we propose our model of organization, which implements Self-Adaptation features using Self-Organization and can improve extracted problems of existing models. Altogether, we have tried to analyze and compare some of main proposed models for Multi-Agent systems' organization. Finally we have proposed a model to improve Self-Organized organizations of Multi-Agent systems.

Keywords: Multi-Agent Systems, Organization, Organization Model, Organization in Multi-Agent Systems, Self-Organization, Self-Adaptation

1. Introduction

Agents are software or hardware units and their general task is to receive a task, process, execute and give the result as an output to a system or another agent. In any organizations, agents get assigned to a role and have particular tasks according to their assigned role. Multi-Agent systems are defined as a combination of agents and societies from their birth. But, typically the focus is on agents and their stations. These are Agent-Oriented Multi-Agent systems. Recently, this focus is on Organization-Oriented Multi-Agent systems, in which concepts like organization, group, relations, performance, and role are more debated [4].

Multi-Agent systems provide a good situation for implementing autonomous systems that can manage themselves [5]. In [11], Multi-Agent systems are defined as an organization of autonomous agents which are trying to reach their common goals. Multi-Agent systems can be used in any field in which concurrent and complex computation is needed. According to Dignum et.al, [7], organizations are the solution for controlling Multi-Agent systems and according to [2], using organizations a group of simple agents can do big and complex jobs and complexity of the agents is reduced. Also, [11] indicates that organization is a social entity that is coordinated informed. This entity has a specific border and acts based on some basic disciplines to reach a particular goal or a set of goals. Because of complexity in agents, numerous agents, and complexity in the system itself, organization in Multi-Agent systems, is distributed and it is not possible to make it centralized.

One of the main problems is that Multi-Agent systems are generally used in uncertain environments. According to Weyns et.al, [6], environment of an agent is anything other than the agent itself and has a role in the system. As a result, other agents are counted as environment for a particular agent. Change is unavoidable in an uncertain environment. Sometimes these changes can cause a situation in which the current structure of organization does not have suitable and required performance and usefulness. Consequently, using permanent rules for an organization would not be applicable. In another words, the systems should be able to adapt to the environmental changes. As a solution, Self-Organization is proposed in [4, 5]. However based on Horling and Lesser [2], Self-Organization always comes along with emergence.

Emergence is the whole behavior of a distributed system that is arisen from behavior of local parts. Emergence is used against reductionism, which stated that a system can be reduced to the sum of its parts. Emergence can cause undesired situations like a Chaos. So, a control on emergence is also needed.

In 2003 IBM proposed an architectural model for Self-Adaptive systems [8].There are four main parts in this architecture: Monitor, Analyze, Plan, and Execute. These four parts also use a knowledge core. According to this model, a system on which this loop is implemented (Managed Element) is always monitored. As changes happen, this loop collects information, analyzes those information, based on knowledge, decides what to do, and executes necessary actions. In this paper we propose a model in which we combine Self-Adaptive features with Self-Organizing features to solve before-mentioned problems. Our model is an organization model of Multi-Agent systems that uses benefits of an overall control to prevent undesired emergence and the same problems.

The rest of the paper is organized as the following: Related works about organizations in Multi-Agent Systems will be introduced and analyzed in section 2. In section 3, these models will be compared to each other and their advantages and disadvantages will be analyzed. Section 4 proposes some ideas and proposals to improve these models and a model to improve all of before-mentioned weaknesses based on the proposed ideas. Finally, conclusion and references are placed in section 5 and section 6, respectively.

2. Related Works

2.1. AALAADIN, a Meta-Model

Ferber et.al, [1] propose a general Meta-Model for Multi-Agent Systems based on organizational concepts like groups, roles, and structures. This model allows utilization of agents with heterogeneous languages, applications, and structures. This paper claims that the key to design and implement complex and ultra large scale systems is to use mentioned organizational concepts as basic concepts in organization and relating them to behavior of the agents. In this view, without considering agents behavioral principles, organization is considered a method to arrange groups and roles to shape a whole entity.

Figure 1. shows basic concepts of AALAADIN model. In this study, no proposals have been made regarding the internal structure of agents and an agent is defined based on its role in an organization. Also, groups are a set of agents and distinguished according to roles they assign to their agents. An agent can be member of more than one group and consequently, can get more than one role. Each role assigned to an agent is local for that group. Based on this paper, there are concepts in defining organization oriented Multi-Agent Systems other than basic concepts (Role, Group, Agents). Figure 2. shows these concepts. In this figure, “Group Structure” entity is an abstraction of group and “Organizational Structure” entity is defined as a set of group structures showing the design of an organization of Multi-Agent Systems.

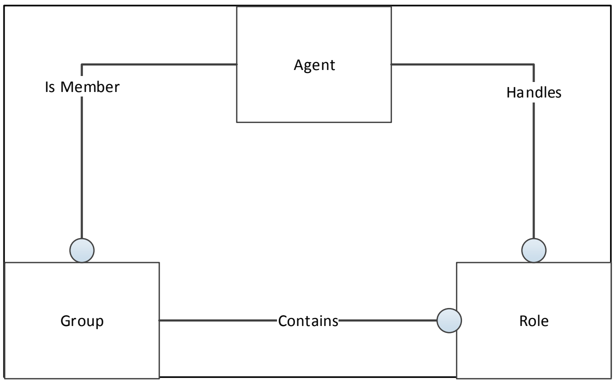


Figure 1. *Conceptual Model of AALAADIN [1]*

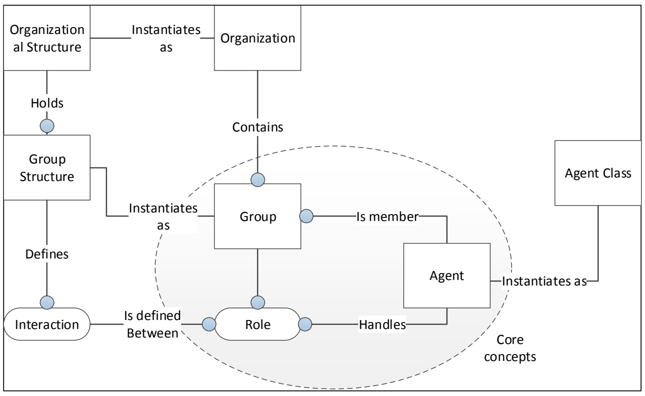


Figure 2. *Methodological Model of AALAADIN [1]*

2.2. Self-Organized Organizations

Kota and et al propose a model for structural adaptation based on Self-Organization in [5]. The method proposed in this paper allows the agents to change their structure of relations to achieve a better distribute of tasks.

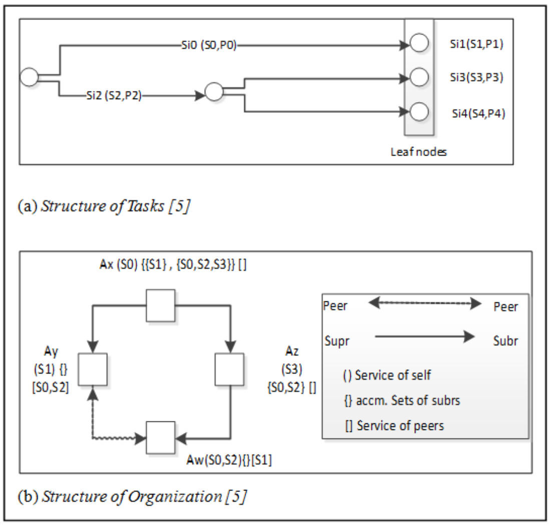
The authors believe that decentralized structural adaptation is the best way to obtain Self-Organization. Structural adaptation consists of change in relations of the agents and as a result, redirection in their interactions. The adaptation method is a continuous and centralized process with participation of all agents to decide how and when to update their relations, only based on local accessible information.

The modeling of organization consists of modeling the agents of the organization, organization features, and task environment. In this model, organization of agents consists of a group of collaborating and problem solving agents located in a task environment. Modeling of a task consists of a set of Service Instances (SIs), duals determining a particular service and the amount of process it needs to be executed placed in a tree data structure. The SIs are executed in a prioritized order and a service is executed when all of its nodes are executed too. In other words, the execution of SIs starts from root of the tree and goes towards leaf nodes. Nodes in the same level are executed in a parallel way. Figure 3.a. shows the tree of SIs.

An organization consists of agents that provide these services. Each agent is a dual consisting set of services it can provide and its computational capability. Figure 3.b. shows the organizational structure. The beginning of a task execution starts with assigning the root of a particular SI tree to a randomly chosen agent. So, each agent has two obligations: Task execution and task allocation. There are three kinds of relation among agents in this model: Acquaintance, Peer, and Superior-Subordinate. The agents would often assign SIs to their subordinates rather than its peers. Since an agent has no idea about tasks that will be provided in the future, the structural adaptation method for Self-Organization in this model uses only historical information of agents. Figure 4. shows different types of reorganization. For example, 1(ii)form-subr means that the relation between two agents changes from acquaintance to superior-subordinate.

2.3. MACODO Middleware

Weyns and et al propose architecture of a middleware for Self-Organization in Multi-Agent Systems called MACODO in [6]. In fact, the MACODO middleware is proposed as a middleware to help organization-oriented designation of Multi-Agent Systems in context-based environments. This middleware omits life cycle management from roles of agents and provides reuse capability and easy understanding, design, and management in Multi-Agent Environments. Therefore, whenever there is a change in the system or environment it would be the middleware that adapts the organization. Figure 5. shows organizational model of MACODO.



(a) *Structure of Tasks [5]*

(b) *Structure of Organization [5]*

Figure 3. *Examples of Task and Organization [5]*

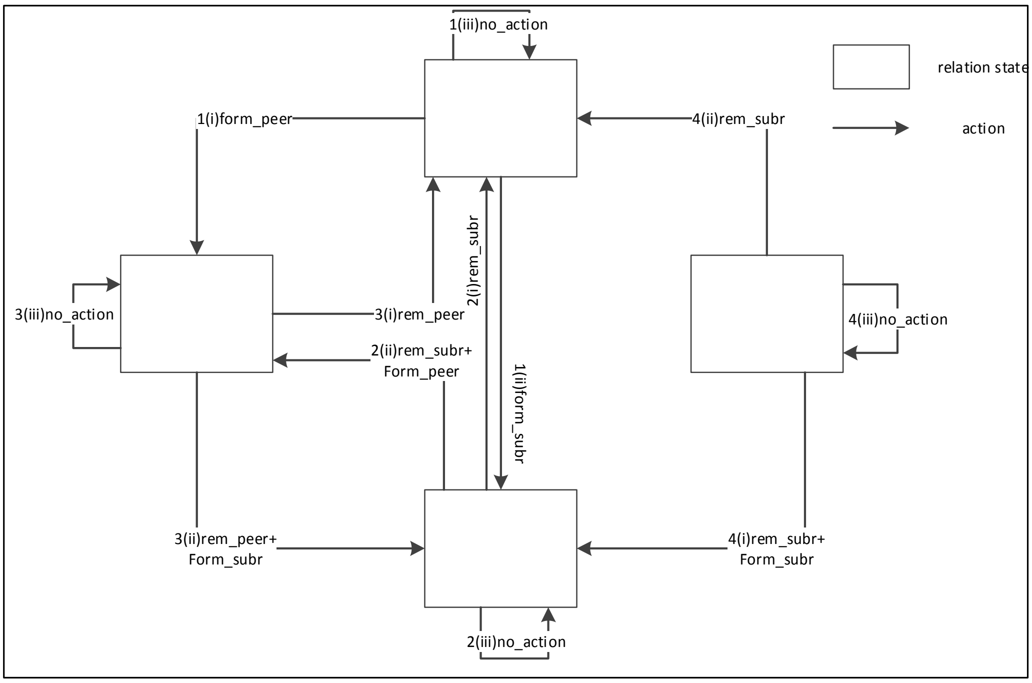


Figure 4. *Graph of Relation Adaptation [5]*

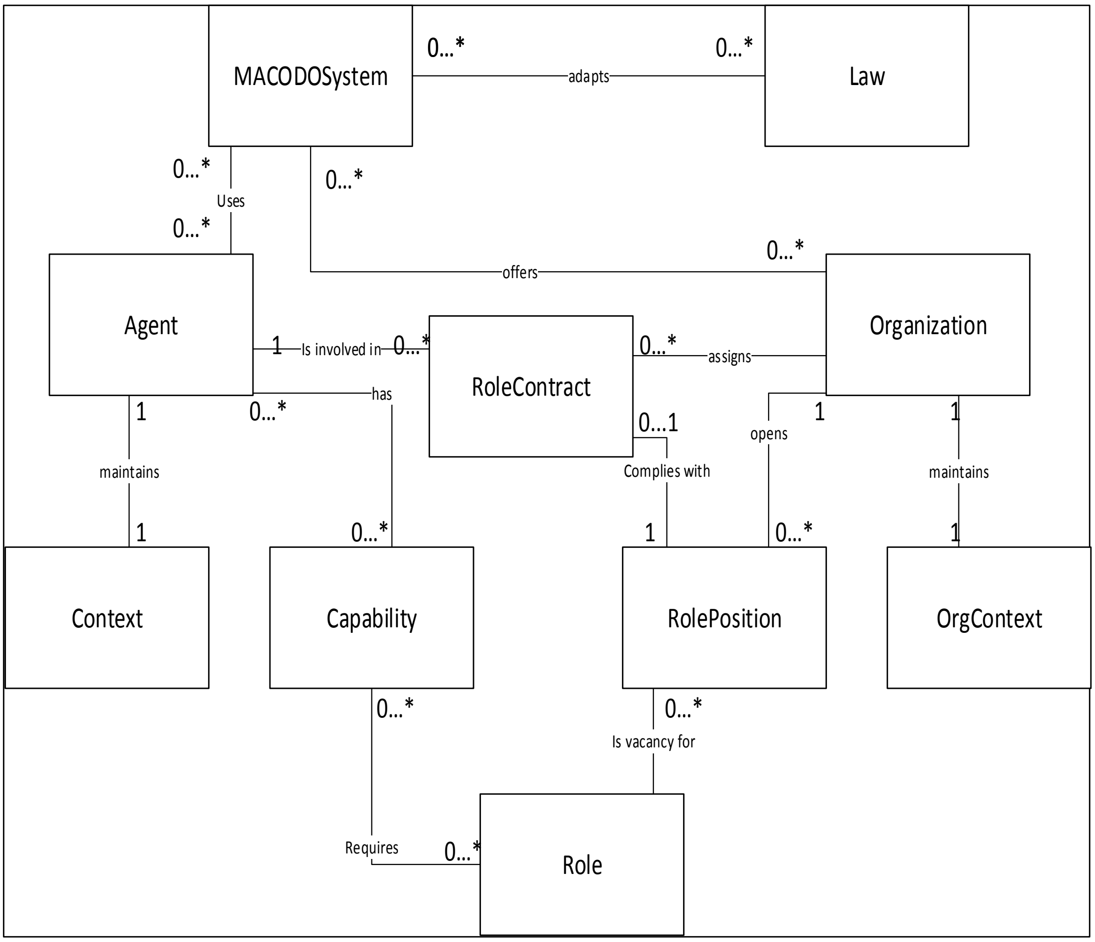


Figure 5. *MACODO Organizational Model [6]*

3. Analysis of models

Among the three, the MACODO middleware is the only model that describes how to implement Self-Organization for Multi-Agent Systems. In this paper, software architecture of agents is proposed and practically tested in a case study about traffic control cameras as agents in roads. Details of advantages and disadvantages of each model are listed in table 1.

Table 1. *Advantages and Disadvantages of Existing Models*

Model name	*Advantages*	*Disadvantages*
AALAADIN Model	- Possibility to use heterogeneous agents - Possibility to have more applications for MAS	- No practical proof of efficiency - No internal structure for agents No method to shape groups and organizations and how to adapt
Self-Organizing Organizations	- Agent-oriented and decentralized Self-Organization and - Detailed solution for adaptation - Practical proof of efficiency	- No internal structure for agents - No real implementation in real world
MACODO Middleware	-Practical and detailed solution for adaptation - Internal structure for agents and how they should connect - Real world implementation - Possibility to use heterogeneous agents - Reusability, easy understanding, better design and management	- No more real world implementations

4. Proposed Model for Organization of Self-Organized Multi-Agent Systems

4.1. Overall Proposals

Having no practical implementation in real world mentioned as one of the weaknesses of introduced models in the previous section. To implement a software, the architecture of that software should be available. As a result, a model for Self-Organizing organizations should be architectural. It means, the model should consider system requirements, propose designation, and make implementation and maintenance possible. Therefore, lack of architectural models for Self-Organizing organizations in Multi-Agent Systems can be considered a problem.

Another problem is collaboration of agents that is missing in these models. In Self-Organization for Multi-Agent Systems the direction of orders is bottom-up and agents decide and perform locally. This feature can cause Emergence phenomena in the system. It can be undesired if it is not controlled. An appropriate solution is to reinforce positive emergence or Synergy and decrease and eliminate negative emergence. Therefore, having global coordination and collaboration among the agents whereas they are independent, decide and perform internally, would be a proper solution. This solution can be founded in Self-Adaptation architecture that has a top-down process. In other words, proposing a Self-Adaptive model for Self-Organizing organizations would be the key to control emergence. This model should make Self-Organization process to act in line with main goals of Self-Adaptation process. As a result, it is necessary to propose a view of collaboration among the agents.

4.2. The Necessity of Architecture for Autonomous Systems

According to Garlan and Schmerl [12] and Cheng et.al, [13], while developing Self-Adaptive systems generally, it is necessary to use architecture of system when adding Self-* features. In these research studies, the reasons that we need architecture to develop Self-Adaptive Systems are expressed as below:

- *Isolated concerns:* applying architecture to Self-Adaptive Systems, we can develop Self-Adaptive features in separated and specific areas.
- *System extensibility:* applying architecture to software development, we can easily add more operations to system.
- *Easier maintenance and evaluation:* since using architecture clarifies area of problem and operation, maintenance and evaluation of Self-Adaptive System will be much simpler.
- *Reusability of components, especially Self-Adaptive ones:* if an appropriate level of abstraction is applied on Self-Adaptive components in an architecture based system, they can be easily utilized in other systems with minimum modification.
- *Possibility of using other systems' components in architecture based Self-Adaptive Systems:* if the components are developed based on specific standards, they can be utilized in Self-Adaptive Systems using a simple interpreter.

4.3. The Proposed Model

As mentioned before, one of the problems of existing models for Self-Organization in Multi-Agent systems is that they are not architectural. Then, we discussed the necessity of architecture for autonomous systems. MAPE-K loop on the other hand, is an existing architectural model for autonomous control on an element. In this paper, we say that the MAPE-K loop architecture can be used to control Multi-Agent systems. Figure 6. shows MAPE-K architecture.

It is true that MAPE-K loop has a centralized control, and Multi-Agent systems are distributed. However, our proposal is not just to use the MAPE-K loop on Multi-Agent systems. In our proposed model, Self-Adaptation, implemented by the MAPE-K loop, determines overall policies, and Self-Organization practically executes the decisions made by the MAPE-K loop. In fact, this is a combination of Self-Adaptation and Self-Organization. In other words, Self-Adaptation monitors the whole system, analyzes the changes, decides what to do, and in execution part, Self-Organization is responsible for execution of decisions made by Self-Adaptation. However, these decisions are about maintenance of the whole system. It means agents are still locally deciding about how to act to emerge to the final goal of the system. Nevertheless, the organization model of Self-Organized Multi-Agent system should support this combination.

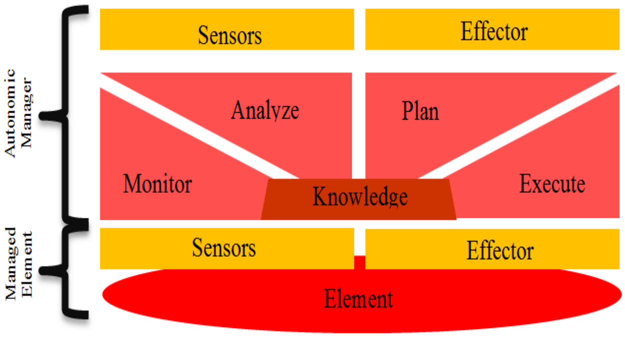


Figure 6. *MAPE-K Control loop [8]*

Horling and Lesser [2] introduce and categorize organizational paradigms for Multi-Agent systems. These paradigms include:

- *Hierarchies:* In Hierarchies, agents are arranged in a structure like trees. It means, the view of agents with a higher level in the tree is more global than that of others in lower levels.
- *Holarchies:* Holons define characteristic of holarchies. Holons can include one or more entities and meanwhile, be a member of one or more superordinate holons.
- *Coalitions:* Consider set A is a population of agents. Each subset of A is a potential coalition. Generally, coalitions are goal directed and have a short life-cycle. It means, coalitions are formed for a specific goal and disappear whenever the goal is reached.
- *Teams:* A common goal is what keeps agents of a team together. In a team, agents work together to reach that common goal. These features make teams and coalitions similar. But, teams try to reach maximum utility of the whole team rather than that of each member. This is what distinguishes teams than coalitions.
- *Congregations:* Like teams and coalitions, agents in congregations work together in a flat organization to reach goals and benefits. The difference is that congregations are not short-lived and formed to reach

multiple goals. They are formed among agents with similar or complementary agents to find more appropriate collaborators.

- *Society:* Societies spontaneously bring a long-lived social collaboration into mind. Societies of agents are fundamentally open systems. It means, agents can join and leave a society if they want. The society acts as an environment in which agents act and interact.
- *Federations:* Federations of agents have many different varieties. Agents of a federation share the common characteristics of a group. They entrust some amount of autonomy to a delegate agent that represents their group. The delegate can have different roles. For example, the delegate can act as an interpreter between agents inside and outside the group, assign tasks to agents inside the group, or, monitor the promotion of other agents.
- *Markets:* In market places, some agents may be buying and some others, seller agents. Buying agents announce their request for some items like resources, tasks, services and goods. On the other hand, selling agents may supply these goods. Sellers or a third party called auctioneer is responsible of processing bids and announcing the winner.
- *Matrix Organizations:* in a strict hierarchical organization, the structure is like a tree in which an agent or a group of agents report to one manager that provides these agents with goals, direction and feedback. This restriction is relaxed in matrix organizational paradigm. In this paradigm, any agent can have multiple managers. Therefore, successful agents can affect multiple entities by their local actions.

From different Organization Paradigms for Multi-Agent systems mentioned above, a combination of federations and hierarchies seems to be the most appropriate paradigm for our model. Figure 7. shows hierarchies and federation paradigm.

Collaboration between agents for coordinating about maintenance and control of the whole system seems to be a good solution for mentioned problems. From paradigms introduced in this part, the most suitable paradigm is the Federation. It means we can choose an agent in each organization to be a representative of the group and transport the information of its own group to other agents. These information can be gathered and transited to an agent in which we have established the so-called MAPE-K loop.

But there is another problem here. What if the agent with MAPE-K roll fails? This architecture makes that specific agent a bottleneck. To solve this problem, we came up with the idea of hierarchical federation. In this model of organization, there is a representative for each organization and these agents can make another organization in a higher level and have another representative.

But, how can this organizational model established in a Multi-Agent system? As you can see in [6], Weyns and et al propose a middleware in which the internal architecture of agents are defined. The agents are responsible for shaping our organizational model, too. Therefore, in agents using our model a component should be defined that is responsible for roll assignment. There are different reference architectures for Multi-Agents systems:

- OMG's Model: In this model agents are categorized based on their capabilities and organizations.
- FIPA's Model: This model is a foundation for physical intelligent agents and aims pragmatic physical intelligent agents.
- KAoS' Model: This model is for standardization of Multi-Agent systems based on knowledge based and agent oriented view.
- General Magic's Model: This model is a commercial model for Multi-Agent systems in communications area.

We avoid more elaboration of these models. Among these reference models, the FIPA's Model is reference of other models and contains details of inside components of an agent. We choose this model for implementation of agents and the whole Multi-Agent system.

Figure 8. show a hypothetical situation of a Multi-Agent system which is organized using our organizational model. It is interesting to know why same agents are in different levels. For example agent B1 is in level 0 and level 1. Being in a hierarchical federation organization is just a roll and as we know, different rolls can be assigned to one agent. So, agent B1 is in level 0 and level 1 and agent A12 is in level 0, 1, and 2. Also, Figure 8.b. shows the practical situation of a system that is implemented using our model. As you can see, the positions of agents is not changed and the roll of being a representative or being a member of a level is just assigned to agents.

Now, we know that decision about being on which level or organization is made by agents, themselves. This decisions and similar ones can be made based on some specific restrictions like environmental conditions.

5. Discussion and Conclusion

In this paper we analyzed some of the relevant state-of-the-art models for organizations in Multi-Agent Systems. In fact, we studied the ones in which organizations are the control tools of Multi-Agent Systems.

In section 2, the top three research studies in this field were elaborated and analyzed. First of all, AALAADIN model, in which a model for Multi-Agent Systems with heterogeneous agents is proposed, was studied and analyzed. Afterward, Self-Organizing organizations model, in which agents can adapt their structural relations to get a better specification of tasks, is introduced and explained. Finally, the MACODO middleware, in which an architectural model is proposed to design and implement Self-Organization in Multi-Agent Systems, was explained and analyzed. After analyzing and comparing these existing models, we extracted some of their advantages and disadvantages. In the next stage, we stated their problems that the main one was lack of architecture. We showed the necessity of architecture for autonomous systems and finally we propose a model in which architecture is considered and the problems mentioned in section 4.1. are also solved.

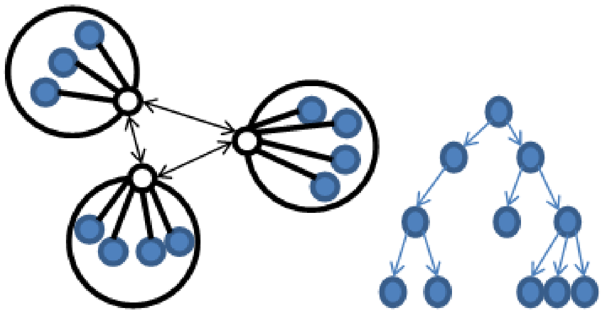


Figure 7. *Federation and hierarchical organizational paradigm [2]*

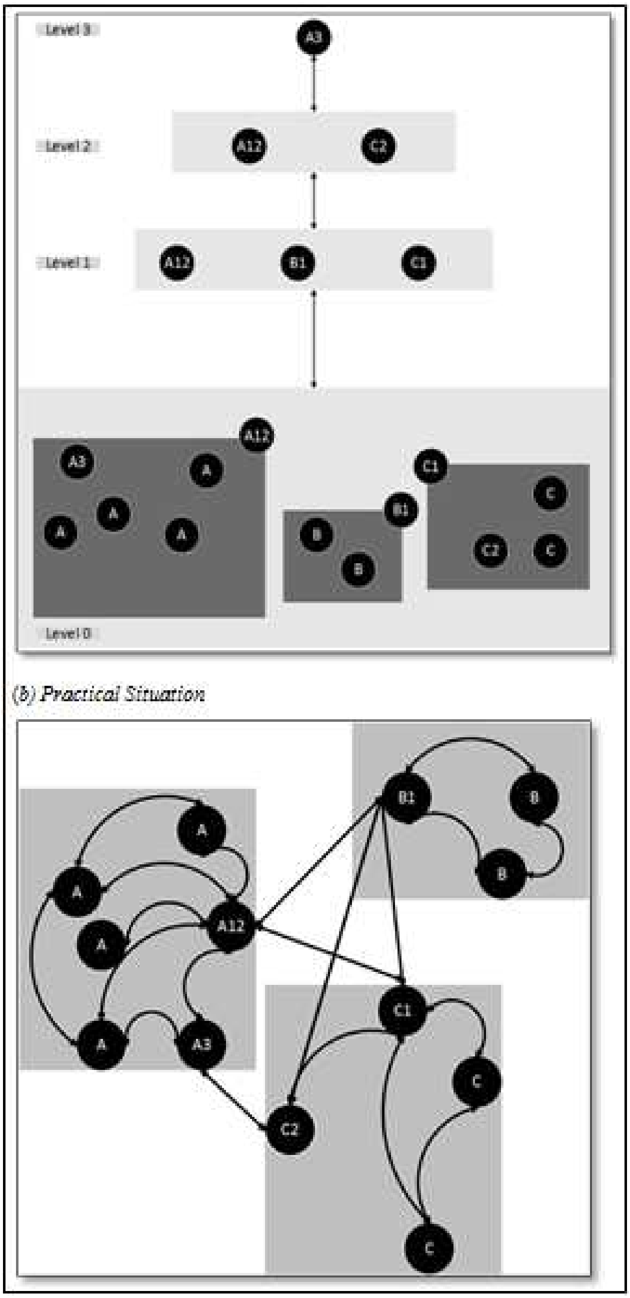


Figure 8. *Proposal Organizational Model*

Our new model combines Self-Adaptation features with Self-Organization features. This way, Self-Adaptation control loop controls the whole system without making it centralized. Self-Organization is the tool of adaptation. Our future work will consist of two main parts: first we should elaborate our model of agents and organization based on FIPA’s reference model, and second, we should try to present more details on the restrictions that agents should make decisions based on.

References

[1] J. Ferber and Gutknecht, “A meta-model for the analysis and design of organizations in multi-agent systems”, International Conference on Multi agent systems, pp. 128-135, 1998, France.

[2] B. Horling and V. Lesser, “A Survey of Multi-Agent Organizational Paradigms”, The Knowledge Engineering Review, Vol. 19, Iss. 4, 2004, pp. 281-316.

[3] K. Y. Rafi, A. Farahani, and E. Nazemi, “An organizational model for autonomic intelligent distributed systems”, 2nd World Conference on Information Technology, pp. 624-630, 2012, Turkey.

[4] J. Ferber, O. Gutknecht, and F. Michel, “From Agents to Organizations: an Organizational View of Multi-Agent Systems”, 4th InternationalWorkshop, AOSE, pp. 214-230, 2004, Australia.

[5] R. Kota, N. Gibbins, and N. R. Jennings, “Self-Organising Agent Organisations”, The 8th International Conference on Autonomous Agents and Multiagent Systems, pp. 797-804, 2009, USA.

[6] D. Weyns, R. Haesevoets, A. Helleboogh, T. Holvoet, and W. Joosen, “The MACODO Middleware for Context-Driven Dynamic Agent Organizations”, ACM Transactions on Autonomous and Adaptive Systems (TAAS), Vol. 5, Iss. 1, 2010, pp. 132-160.

[7] V. Dignum, F. Dignum, L. Sonenberg, “Towards Dynamic Reorganization of Agent Societies”, Workshop on Coordination in Emergent Agent Societies, pp. 32-39, 2004, Spain

[8] An architectural blueprint for autonomic computing, IBM Group, 2003.

[9] W. Truszkowski, M. Hinchey, J. Rash, and C. Rouff, “NASA’s Swarm Missions: The Challenge of Building Autonomous Software”, IT Professionals, Vol. 6, Iss. 5, 2004, pp. 47-52.

[10] P. Mathieu, J. C. Routier, and Y. Secq, “Dynamic Organization of MultiAgent Systems”, second international joint conference on Autonomous agents and multiagent systems, pp. 451-452, 2002, USA.

[11] M. Kolp, P. Giorgini, and J. Mylopoulos, “Multi-Agent Architectures as Organizational Structures”, Autonomous Agents and Multi-Agent Systems, Vol. 13, Iss. 1, pp. 3-25, 2006

[12] D. Garlan and B. Schmerl, “Model-based adaptation for self-healing systems,” Proceedings of the first workshop on Self-healing systems - WOSS ’02, p. 27, 2002.

[13] S. Cheng, D. Garlan, and B. Schmerl, “Software Architecture-based Adaptation for Pervasive Systems” International Conference on Architecture of Computing Systems Karlsruhe, pp. 67-82, Germany, April 8–12, 2002

[14] M. Risoldi, J. L. Fernandez-Marquez, G. Di Marzo Serugendo, "Resilience Framework for self-organising systems", Adaptive, Dynamic, and Resilient Systems, G. Cabri and N. Suri (Eds), Taylor and Francis, 2013.

[15] G. Di Marzo Serugendo, J. L. Fernandez-Marquez, "Self-Organising Services", Int. Conf. on Self-Adaptive and Self-Organizing System, Philadelphia, USA, 2013.

[16] G. Stevenson, D. Pianini, S. Montagna, M. Viroli, J.Ye, S. Dobson, "Combining self-organisation, context-awareness and semantic reasoning: the case of resource discovery in opportunistic networks", Proceedings of the 28th Annual ACM Symposium on Applied Computing, Coimbra (POR) March, 2013.