Supporting Autonomy in Agent Oriented Methodologies

Valeria Seidita[1,2] and Massimo Cossentino[2]

[1] Dip. di Ingegneria Chimica, Gestionale, Informatica, Meccanica
University of Palermo, Italy
valeria.seidita@unipa.it
[2] Istituto di Reti e Calcolo ad Alte Prestazioni
Consiglio Nazionale delle Ricerche, Palermo, Italy
cossentino@pa.icar.cnr.it

Abstract. *Designing a software solution for a complex systems is always a demanding task, it becomes much more complex if we consider to design a multi agent system where agents have to exhibit autonomy; which abstractions and which concepts to take into consideration when using a design methodology we would like to support autonomy? In this paper, we answer this question by studying and analyzing literature on the concept of agents in order to establish the basic set of concepts an agent oriented methodology has to deal with.*

Keywords: Autonomy, Multiagent systems, Design Process.

1 Introduction

So far, researchers have faced the problem of understanding, from the point of view of multiagent system (MAS) development and implementation, how to construct interacting or communicating agents, how to let agents reason, how to ensure they take useful decisions and so on. Several good solutions, in terms of techniques and methods, have been identified but however we find they mainly focus on an implementation perspective and on the features agents must possess, not how these features have to be abstracted for providing means for carrying on good analysis phases.This fact does not create so many problems when dealing with concrete features, like for instance goal, but become much more awkward when dealing with non tangible features, like for instance *autonomy*.

What we want to do now is moving the problem towards the design process and identifying which abstractions do have to, or do not, be present in the analysis phase of an agent oriented methodology for supporting agent autonomy.

The objective of this work is to explore agents features in order to identify which elements (or abstraction) an agent based methodology have to deal with for being countered among the methodologies implementing autonomy. In doing this, in the next section, we explore building definitions of agent, multiagent system and agent based methodology; among the three, *agent* is the concept that meets smaller consensus among researchers, however this does not affect

our analysis indeed we explore literature on agent definition from the point of view of the designer that has to identify and highlight which elements he has to deal with mainly during analysis and then during the design in order to implement agents to be useful for solving real world problems. For doing this we start from the features an agent has to own and we report some of the most popular definitions from the plainest to the most structured.

Within the end of section 3 we provide a reasonable list of elements that have to be present in a methodology for supporting autonomy design whereas in section 4 we show two agent oriented methodologies supporting autonomy in step with our evidences. Finally, some discussions and conclusions are provided.

2 Agents Definitions

In this section we overview literature in order to examine the notion of agent and autonomy in agency.

Russell and Norvig define the agent as follows:

> *"An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors"* [15].

In this sense, we may venture that everything that uses inputs from an environment and provides output may be considered an agent.

According to Maes:

> *"Autonomous agents are computational systems that inhabit some complex dynamic environment, sense and act autonomously in this environment, and by doing so realize a set of goals or tasks for which they are designed"*[13][14].

Maes directly defines autonomous agents, she does not separate the term agent from the term autonomous, they are tightly interrelated, and, in addition to Russel and Norvig's idea, autonomy is also in the action and sensing and a new important element is considered, the goal. This definition is more restrictive, it is not sufficient to act and sense but autonomous agent has to autonomously act and sense in a dynamic environment in order to pursue its own goals.

Haes-Roth defines:

> *"Intelligent agents continuously perform three functions: perception of dynamic conditions in the environment; action to affect conditions in the environment, and reasoning to interpret perceptions, solve problems, draw inferences, and determine actions."* [11]

Hayes adds for agent the reasoning and introduces the concept of affecting the environment with actions and determining which action to perform.

In a white paper by IBM:

> *"Intelligent agents are software entities that carry out some set of operations on behalf of a user or another program with some degree of independence or autonomy, and in so doing, employ some knowledge or representation of the user's goals or desires."* [10]

The definition of IBM instills the presence of users and the fact that the agent acts on the behalf of them, and also gives suggestion for understanding autonomy by means of the word "independence". Moreover, the need for user's goals and desires representation arises.

Brustoloni gives the following definition of autonomous agents.

> *"Autonomous agents are systems capable of autonomous, purposeful action in the real world."*[2]

A very concise definition that highlights three elements, the ability of acting, the environment rather the real world, the presence of driven or committed behavior suggested by the word *purposeful*.

Franklin and Graesser state that:

> *An autonomous agent is a system situated within and a part of an environment that senses that environment and acts on it, over time, in pursuit of its own agenda and so as to effect what it senses in the future* [9].

Franklin and Graesser embrace all the previous definitions and restate the fact that the environment is continuously changing also for the effect of agents' actions.
It is worth to note that in all these definitions the concept of reaction to the environment changes seems to be hidden in the concept of action.

One of Wooldridge's definitions of agent reports that:

> *"An agent is a computer system that is situated in some environment, and that is capable of autonomous action in this environment in order to meet its delegated objectives."*[17]

The previous definition summarizes the more complete one from [16][17], it is the *weak notion of agent* for Wooldridge and Jennings but however complete enough for the purposes of this paper:

> *An agent is a system enjoying the following properties:*
> - *autonomy: agents encapsulate some state (that is not accessible to other agents), and make decisions about what to do based on this state, without the direct intervention of humans or others;*
> - *reactivity: agents are situated in an environment,(which may be the physical world, a user via a graphical user interface, a collection of other agents, the INTERNET, or perhaps many of these combined), are able to perceive this environment (through the use of potentially imperfect sensors), and are able to respond in a timely fashion to changes that occur in it;*

- *pro-activeness: agents do not simply act in response to their environment, they are able to exhibit goal-directed behavior by taking the initiative;*
- *social ability: agents interact with other agents (and possibly humans) via some kind of agent-communication language, and typically have the ability to engage in social activities (such as cooperative problem solving or negotiation) in order to achieve their goals.*

So, Wooldridge, too, highlights the goal directed behavior of agents in an environment that changes while the procedures for pursuing goals are running. This leads to a kind of reactivity that is different from the one of the previous definitions, the agent has to react in a timely fashion to the changes. Agents reach autonomy if there is not the intervention of humans and if they are engaged in social activities.

Moreover, in [17], Wooldridge says *"An agent takes sensory input from the environment, and produces as output, actions that affect it. The interaction is usually an ongoing, non-terminating one."* An agent continuously looks at the environment and gains information from it, on the basis of this information it decides what to do next, what action to perform next in order to pursue its own agenda or goals, each action typically affects and changes the environment; once acting the agent senses again the environment and so on in a continuos loop *sense-decide-act-sense-decide*. During this loop, decisions depend on agent's (mental) state, it will take different decisions basing on the particular interaction with the environment (hence its own state and the environment state at a specific time).

In our opinion, one of the most comprehensive definition found in literature is the one given by Jacques Ferber in [8]:

"An agent is a physical or virtual entity:
- *which is capable of acting in and perceiving its environment, can communicate directly in an environment and possesses resources of its own;*
- *which has only a partial representation of this environment (and perhaps none at all), is driven by a set of tendencies (in the form of individual objectives) and possesses skills and can offer services;*
- *whose behavior tends towards satisfying its objectives, taking account of the resources and skills available to it and depending on its perception, its representation and te communication it receives."*

Ferber's definition points out that agent is able of acting and not only reasoning (different viewpoint from AI field), by simply acting agent modifies its environment and its future interaction with it and its decision making process. An agent communicates with other agents that are part of the environment (hence environment includes not only the static object but also other autonomous agents). Finally, autonomy is meant in the sense that agents are not directed by commands from users, but by a set of tendencies, i.e. goals or some other kind of desire the agent wants to realize.

In the following section we explain how all these definitions, which we know contribute in a different way to the concepts of agent and autonomy, led us to identify the elements an agent design methodology has to present in order to support autonomy.

3 Towards implementing autonomy in agent oriented methodologies

Several agent design methodologies are reported in literature and are used for solving different problems (see [7][1][12] for a wide overview), but how may we affirm that a design methodology supports analysis and design of autonomous agents given that, as we have already seen, the concept of agent and of autonomy in a multiagent system includes a variety of aspects to be taken into consideration and that are sometimes not coherent?

Besides, too many methodologies force us to think in terms of objects whereas for an agent oriented methodology we have to think in terms of agents being led by the sentence "on the behalf of humans" seen in some previous definitions. When we perform object oriented design we tend to represent, to model, the users' interactions with the system (with the software objects representing or implementing the objects in the real world) in a functional way, hence conducting analysis on the base that some inputs have to be elaborated to give some outputs. When we perform agent oriented design (it is ascertained that objects and agents are truly different) we have to think of the agents of the system and to model the software under the point of view of agents, of their interaction with the world and with other agents that populate the world; at the same time we have to think that environment changes and all this is away from the classical software engineering perspective of functional design. But what does it mean and imply? From all the definitions of the previous section we may abstract that: an agent is a purposefully originator of actions; agent's actions are chosen in the interest of the agent itself; agent senses the environment; actions modify or shape the environment; agent takes decisions in an independent fashion on the actions it may do. Let us now further analyze these sentences with the aid of the Table 1.

From a methodological point of view, a designer has to analyze what an agent has to do in order to accomplish the user design objectives hence its goals, a list of actions it is able to do and a list of actions it is not allowed to do; the first depends on the kind of agent situated in an environment whereas the second on the kind of environment it lives in; the environment includes other agents. In other words, it is important to analyze and to establish the boundaries within which agents may act, thus not deciding how it behaves but only modeling agent decision process that is realized by dynamic, not fixed, plans for pursuing goals and may be influenced by several factors (an important one is the set of non-functional requirements of the systems, we don't detail this in the present paper).

From the study of the autonomy definitions and the survey of many methodologies, we identified the minimum set of elements a design methodology has to

Table 1. Design abstractions

Agent autonomy feature	**Consequences in the design abstractions**
"agent is purposefully originator of actions"	an agent has to be endowed with *knowledge* on its agenda and *goals* and on the *action* it may perform **or not** in order to pursue them
"actions are chosen on the interest of agent"	agent has to be aware of the action it is allowed (*constraint*) to perform in order to pursue an objective
"agent senses the environment"	agent has to be endowed with *knowledge* about its *environment*
"actions modify or shape the environment"	there is a relation between *action* and *environment*, the list of actions an agent may do are related to its environment
"agent takes decision in an independent fashion on the actions"	there must not be someone who says to an agent what it has to do, hence *no static plan* but autonomous *decision process*

encompass (see Fig. 1). In the following section we try to examine two methodologies on the base of these three levels in order to say if they support autonomy.

4 Analyzing how Gaia and ASPECS support autonomy

In this section we use the results illustrated in section 3 and in Fig. 1 in order to examine two well know agent oriented methodologies and to give an example of how to support autonomy.

Gaia [19][3][18][7] is a complete agent oriented methodology allowing the designer to develop multiagent systems going from analysis to code and, so as a lot of other methodologies in literature, considers the requirement elicitation as an independent phase. During analysis phase Gaia moves from abstract concepts to concrete ones creating models detailed in an incremental fashion. Gaia, in its latest version [19], focuses on organizational abstraction in order to analyze and design MASs that work in complex and open environment.

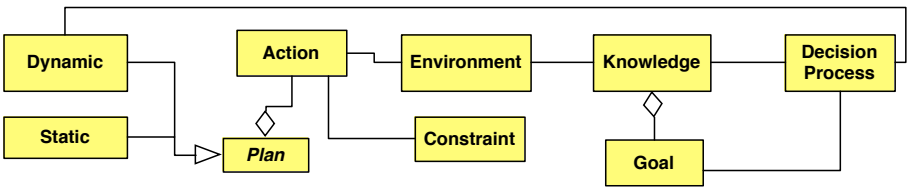

Fig. 1. The minimum set of design abstractions for autonomy.

Gaia is composed of three main phases: analysis, architectural design and detailed design; the first phase is mainly concerned with the identification of organizations of the MAS and their roles, the second phase deals with organizational structure and the third specifies the AgentType concept for modeling what agent plays one or more roles and all the activities, agents may perform, in terms of services (see the provided references for a deeper review of Gaia).

Table 2. Autonomy supported in Gaia

	GAIA
environment	In GAIA the environment is modeled by determining all the entities and resources that MAS may exploit; the environment is considered both physical and virtual so providing a complete and wide representation of all the entities, agents or not, living in it.
action	Entities and resources identified in the environment model are used for modeling what actions according to specific permission may be performed by roles.
knowledge	Gaia provides knowledge on the resources of the environment described in terms of symbolic names (associated with the type of action an agent may do on it); also, knowledge on how to achieve goals by means of *skills*
goal	Goals are identified and modeled during the analysis phase, Gaia principally focusses on the goals of organizations, one or more agents into an organization accomplish goals.
constraints	Rules expressing constraints the organization has to obey in its behavior while executing an activity. In particular during the *Identify Protocol Dependencies* and *Identify Environmental Constraints*, Gaia provides norms and constraints for the interaction among roles and with the environment.
plan	There are not static plan, and this is favorable for well supporting autonomy, and how to take decisions is helped by the identification of Safety and Liveness properties, the first guarantees to prevent undesirable behavior whereas on the contrary the second enables desirable behavior.
decision process	The identification of Responsibility together with Safety and Liveness properties model the decision process in terms of expected behavior of a Role.

ASPECS [6][4][5] is an agent design methodology for developing Holonic MASs; it covers all the activities from requirement analysis to code and allows to design open, dynamic and complex systems. ASPECS combines the holonic and agency concepts for completely modeling the whole structured organizational aspect of entities composing a complex system, thus an organizational approach is rife in all the methodology.
ASPECS is composed of three main phases: System Requirements, Agent Society and Implementation and Deployment. The first phase deals with the definition

of system requirements and the identification of organizations, the second handles the definition of roles, communications, agents and the holonic architecture and finally during the third phase a solution is complemented by using platform dependent concepts.

In Table 2 and in Table 3 we illustrate how each abstraction of Fig. 1 is dealt with in Gaia and in ASPECS. Both Gaia and ASPECS support autonomy,

Table 3. Autonomy supported in ASPECS

	ASPECS
environment	A representation of the environment may be extracted from two different perspectives, the knowledge domain represented through the ontology and the set of resources that may be manipulated by roles through capacities.
action	During the System Analysis phase the concept of *capacity* is used for describing the "know-how" of each agent, a sort of behavioral building block and also the specification of the system and environment transformation under certain constraints.
knowledge	An ontology is used for conceptualizing knowledge domain, hence concepts/objects of the world, assertions on concepts using predicates, and actions an entity may perform for changing properties of one or more concepts.
goal	ASPECS models individual and collective goals, the first kind relates to the self-interest goal of each agent, whereas the second to the goals shared among other agents. Goals are identified during the design phase.
constraints	Social rules and norms are represented in the ontology and are used by roles that represent expected behavior and a set of rights and permissions
plan	Both static and dynamic plans are modeled, the second through Interactions that is a not a priori known sequence of events that may trigger effects on the system.
decision process	It is modeled in the recruitment process that allows to establish and assigning capacities to a holon.

indeed for both of them we found ways of dealing with the abstractions in the table. Gaia explicates knowledge on both actions and environment also providing a model for this whereas ASPECS does not specifically model environment; knowledge of the domain and on the allowed actions is supplied respectively by the ontology model and capacity description. The concept of capacity is very important because different aspects of autonomy are supplied, however they are not clearly made explicit as it is in Gaia. It is not in the scope of this paper to compare methodologies or provide metrics for autonomy, we used these two methodologies for showing that, although deeply different, they both support autonomy; if only one of the above concepts missed we could not affirm the same.

5 Discussion and Conclusions

Agent based systems technology is a paradigm born for conceiving, analyzing, designing and implementing complex systems by means of a very powerful abstraction: *agent*. It is recognized that the most of today's complex systems may be managed through organized societies of agents that communicate and interact in order to pursue their own goals or the ones of the society they live in. In doing this, agents strongly interact with their environments.

All the researchers, working in this field, cannot disagree that using the agent paradigm goes beyond providing a system with problem solving or social interaction capabilities; there must be much more. Autonomous behavior, hence *autonomy*, of agents or of the whole multiagent system is a key concept in agency. So we wondered: since an agent is mostly used as a design paradigm, how may we say that an agent methodology supports autonomy in building MASs? In order to answer these questions we firstly analyzed literature definitions of agent and we conducted a study for catching all the elements that could be used as design abstractions for guaranteeing autonomy.
The result was that, for being autonomous, an agent has to be endowed with the ability to be reactive towards the environment and then to have a representation of it; reactive in the sense that, while pursuing their objectives, agents (living) in a dynamic and uncertain environment change the environment itself and react to that in a useful fashion applying decision process and not static plans imposed by the designers. Moreover, an agent has to be provided of knowledge on the actions it can do, all its abilities, and the rules (or constraints) preventing it to do what is not allowed in a specific environment while pursuing objectives.

Thus the minimum set of elements to be managed during (mainly) analysis phase is: environment, action, knowledge, goal, constraint, decision process and plan; this latter has to be dynamically created by a decision process for realizing autonomy.

This representation, however, does not consider the fact that, see Wooldridge's definition, agents interact with other agents in order to achieve their goals, hence they are engaged in social actions; indeed, generally speaking, few goals may be achieved without the interaction with other people, without getting organized with others, negotiating and cooperating. Thus, we may say, the different levels of autonomy in realizing complex function pass through establishing social actions and creating organizations. Discussing this is out of the scope of this paper and will be argued in the future.

The paper concludes with an example of how two well known agent methodologies support autonomy; this work does not want to be a framework for comparing methodologies on the autonomy aspects, but is a starting point for conducting studies for identifying which are the best practices for constructing methodologies, or for modifying existing ones, well supporting all the aspects of agency, from autonomy to pro-activeness and social ability. So in the future we plan to extend this study to all other peculiar aspects of agency.

References

1. Federico Bergenti, Marie-Pierre Gleizes, and Franco Zambonelli. *Methodologies and software engineering for agent systems: the agent-oriented software engineering handbook*, volume 11. Springer Science & Business Media, 2006.
2. Jose C Brustoloni. *Autonomous agents: characterization and requirements*. Citeseer, 1991.
3. Luca Cernuzzi, Thomas Juan, Leon Sterling, and Franco Zambonelli. The gaia methodology. In *Methodologies and Software Engineering for Agent Systems*, pages 69–88. Springer, 2004.
4. Massimo Cossentino, Stephane Galland, Nicolas Gaud, Vincent Hilaire, and Abderrafiaa Koukam. An organisational approach to engineer emergence within holarchies. *International Journal of Agent-Oriented Software Engineering*, 4(3):304–329, 2010.
5. Massimo Cossentino, Nicolas Gaud, Vincent Hilaire, Stéphane Galland, and Abderrafiâa Koukam. Aspecs: an agent-oriented software process for engineering complex systems. *Autonomous Agents and Multi-Agent Systems*, 20(2):260–304, 2010.
6. Massimo Cossentino, Vincent Hilaire, Nicolas Gaud, Stephane Galland, and Abderrafiaa Koukam. *Handbook on Agent-Oriented Design Processes*, chapter The ASPECS Process, pages 65–114. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014.
7. Massimo Cossentino, Vincent Hilaire, Ambra Molesini, and Valeria Seidita. *Handbook on Agent-Oriented Design Processes*. Springer, 2014.
8. Jacques Ferber. *Multi-agent systems: an introduction to distributed artificial intelligence*, volume 1. Addison-Wesley Reading, 1999.
9. Stan Franklin and Art Graesser. Is it an agent, or just a program?: A taxonomy for autonomous agents. In *Intelligent agents III agent theories, architectures, and languages*, pages 21–35. Springer, 1996.
10. Don Gilbert. Intelligent agent: The right information at the right time.
11. Barbara Hayes-Roth. An architecture for adaptive intelligent systems. *Artificial Intelligence*, 72(1):329–365, 1995.
12. Brian Henderson-Sellers and Paolo Giorgini. *Agent-oriented methodologies*. Information Science Reference, 2005.
13. Pattie Maes. *Designing autonomous agents: theory and practice from biology to engineering and back*. MIT press, 1990.
14. Pattie Maes. Artificial life meets entertainment: lifelike autonomous agents. *Communications of the ACM*, 38(11):108–114, 1995.
15. Stuart Russell, Peter Norvig, and Artificial Intelligence. A modern approach. *Artificial Intelligence. Prentice-Hall, Egnlewood Cliffs*, 25:27, 1995.
16. M. Wooldridge and N.R. Jennings. Intelligent agents: Theory and practice. *The Knowledge Engineering Review*, 10(2):115–152, 1995.
17. Michael Wooldridge. *An introduction to multiagent systems*. John Wiley & Sons, 2009.
18. Michael Wooldridge, Nicholas R Jennings, and David Kinny. The gaia methodology for agent-oriented analysis and design. *Autonomous Agents and multi-agent systems*, 3(3):285–312, 2000.
19. Franco Zambonelli, Nicholas R Jennings, and Michael Wooldridge. Developing multiagent systems: The gaia methodology. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 12(3):317–370, 2003.