Modeling Agents with the CSP formalism: an approach for Optimization of Distributed Problems

Sana Moujahed[1−2] Olivier Simonin[1]
Abder Koukam[1]
1. Université de Technologie de Belfort-Montbéliard, France
{sana.moujahed}{olivier.simonin}{abder.koukam}@utbm.fr

Khaled Ghédira[2]
2. Ecole Nationale des Sciences de l'Informatique de Tunis, Tunisie
Campus universitaire La Manouba, 2010 Tunisie
khaled.ghedira@isg.rnu.tn

Abstract

Multi-Agent Systems (MAS) are an efficient approach to deal with distributed complex problems. However, they are slightly suited to optimization ones. In order to tackle this problem we present a model to specify Multi-Agent Systems with the Constraint Satisfaction Problem (CSP) formalism. It allows to take advantages from CSP solving algorithms and to deal with optimization problems. For this purpose we specify agents as Constraint Satisfaction and Optimization Problems, i.e. CSOP and consequently the MAS as a Distributed CSOP (DCSOP). We illustrate the interest of our approach on the antennae parameter setting problem which is an example of distributed constrained problem. We implement this specification to validate the optimization function model that we propose.

1. Introduction

The resolution of optimization problems with a multi-agent approach remains arduous due to the difficulty to design both agent architectures and distributed optimization algorithms. This paper proposes an approach to specify *MAS* with the *CSP* formalism, that allows in particular to deal with over-constrained problems and optimization ones (for instance in distributed systems, networks and radiomobile networks).

We claim that combining the *CSP* formalism and the multi-agent approach allows to cope with complex problems by achieving some of the benefits of both *CSP* techniques (centralized and distributed algorithms) and multi-agent models/properties (robustness, flexibility, proactivity, etc).

Last decades witnessed significant researches relating to *MAS* approaches as they provide efficient modeling of distributed complex problems. An important number of agent formalisms and architectures is present in the literature. A non-exhaustive list comprises the subsumption architecture [1], the Belief Desire Intention (*BDI*) architecture [6], the BRIC formalism [3], the Gaia methodology [10], etc. These different approaches for agent specification were proved to be efficient to deal with some aspects of the multi-agent problematics, such as inter-agent communication. It is the case with the *BDI* architecture which has been extended to deal with communication between agents. The *BDI* architecture is also well suited for describing an agent's mental state. Moreover, some architectures are adapted to reactive models as the subsumption one. However, most of these models don't include tools or algorithms for optimization problem solving. At the opposite, the approach we propose consists to express a multi-agent system through a formalism presenting existing resolution techniques, i.e. the *CSP* one. We focus in this paper on the expression of *MAS* within the *CSP* formalism. We name this *MAS* rewriting the specification phase.

The remainder of the paper is organized as follows: section 2 presents the main steps and objectives of our approach. Section 3 presents how the elements of a *MAS* may be specified with the *CSP* formalism. Then in section 4 we illustrate our model on the antennae parameter setting problem, through its specification and its implementation within a multi-agent platform. Finally, the last section is devoted to the conclusion and further research.

2. The specification steps

The Multi-Agent approach is an efficient model to decentralised systems since they are too large to be solved by a single centralized approach. The request we focus on is how to define and engineer *MAS* for optimization problems ?
To deal with this challenge, we propose to use the *CSP* formalism to specify and program multi-agent systems. The primary driving force for this choice is that *CSP* is a powerful and generic formalism to simply represent complex problems.
The key idea of the paper is to define an agent as a *CSOP*, i.e. a Constraint Satisfaction and Optimization Problem and consequently a *MAS* as a *DCSOP*, i.e. a Distributed Constraint Satisfaction and Optimization Problem.
Figure 1 illustrates the proposed approach. The first step precises that we must dispose of a multi-agent model of the problem to deal with (all the elements of the problem must be agentified). We assume that the

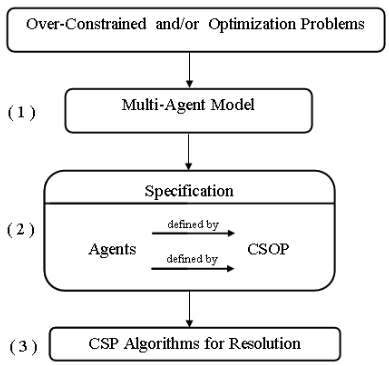

Figure 1. The specification steps

model is given in terms of agents characteristics and their interactions. Note that the agentification of the elements of the problem is not the subject of the paper. Our contribution concerns the formal specification of the agent model via the *CSP* formalism (specification step (2) of figure 1). In our approach, every agent is defined by a *CSOP*, where the objective functions are translated to optimizing functions thanks to the Satisfaction-Altruism model [2] (presented in next section). Consequently the *MAS* is defined by a *DCSOP*. Then, the resulting model allows to use *CSP* based algorithms for the problem solving. Next section describes the second step, i.e. the specification.

3. Modeling a *MAS* as a *DCSOP*

3.1. CSP, CSOP and DCSOP definitions

Let's recall the *CSP*, *CSOP* and *DCSOP* definitions:
A Constraint Satisfaction Problem [9] is the tuple $< X,d,C >$, where X is a set of n variables, $X = x_1, ..., x_n$, $d : X \rightarrow D \subseteq R$ is a mapping between a variable and the domain of its possible real values and C is a collection of m constraints $C = C1, ..., Cm$. Each constraint is a proposition over a subset of the available variables $S \subseteq X$, called the scheme of the constraint. A solution to a *CSP* is an assignment of values that maps all variables in X with a value compatible with d that satisfies all constraints in C.
A Constraint Satisfaction and Optimization Problem is a *CSP* with an associated objectif function *F*, i.e., it is a tuple $< X,d,C,F >$ where $F : S \subseteq X \rightarrow R$. A solution of a *CSOP* is a solution of the underlying *CSP* that maximizes the objective *F*.
Finally, a *DCSOP* is a Distributed *CSOP* in which variables and constraints are distributed among multiple agents. A solution to a *DCSOP* is an instanciation (or several ones) that satisfies all inter-agent/intra-agent constraints and optimizes the global objective function.

3.2. Agent properties

We distinguish for an agent two basic components: the individual agent properties and the interactive properties:
The *individual* agent *properties* are defined by its own abilities, e.g. its resources, intentions, engagements, satisfaction function, memories and experiences, emotions, beliefs, expertise, explicit plans, information on acquaintances, tasks or actions, etc (see [3] for an exhaustive list).
The *interactive properties* are defined by the interactions between agents. We distinguish three forms of interaction: cooperation, coordination and negotiation. They are supported by direct and indirect communication. In this paper we focus on the communication via direct messages exchange. We then identify several types of messages derived from work on speech-act-theory. A generic catalogue was identified [8], it consists of five classes of messages that diverge on the content level and on the effect produced on the recipient agent. We quote *assertive* messages, *directive* messages which involves *interrogative* and *executive* messages, *promissing* messages, *expres-*

sive messages and *declarative* messages. These kinds of messages as well as interaction forms are differently considered in the proposed model. Note that communication in *MAS* implies that the obtained DCSOP presents dynamic aspects: variables, values and constraints can change and evolve.
Since we propose to model an agent by a *CSOP*, we present in the following part the analogy between the two concepts by identifying for an agent what can represent variables, values, constraints and optimization functions.

3.3. The agent's variables

From the previous agent definition, we can infer two classes of variables:

1. Variables induced from the individual properties: every **elementary property** identified previously will henceforth refer to a variable, and will increase the number of variables relative to the *CSOP*.
2. The second type of variables emerges from **inter-agent interactions**, these variables are consequent to **messages exchange**. For example an agent A can advise an agent B to take into account a new variable in its reasoning process. However an agent can infer from its interactions that it has to add by itself a variable to its knowledge.

Formally the set of *CSOP* variables ralating to an agent i at time t is defined as following:

$$V_i = V_{csop} = V_{properties} \cup V_{interactions} \tag{1}$$

$$V_{Properties} = \{V_{P1}, V_{P2}, ..., V_{Pn}\} \tag{2}$$

Where V_{Pi} represents an individual property among the assortment of properties already identified.

$$V_{interactions} = \{V_{I1}, V_{I2}, ..., V_{Im}\} \tag{3}$$

Where V_{Ii} represents a variable generated from inter-agent interactions (communication).

3.4. The agent's values

The domain of variable values, can be divided in two sub-domains:

- The first one corresponds to the initial values (i.e. in the initial state of the system).
- The second one is constructed dynamically from interactions.

This means that every type of message will bring new values to variable domains and can also suppress values or also modify some variable values (except directive ones that will be interpreted as inter-agent constraints, see section 3.3).
Formally, we can set the values domain of variable as the union of two sub-domains:

- $Domain_{initial}$: the agent's initial values.
- $Domain_{interactions}$: values brought from messages.

Thus for the variable j we have:

$$Dj = Dj_{initial} \cup Dj_{interactions} \tag{4}$$

3.5. The agent's constraints

In the *CSP* formalism, the constraints concern the variable set, they represent restrictions on the set of values that can be taken by a variable. Various implementations are possible for constraints: in intention by functions, arithmetic inequalities, etc, or in extension by the authorized values. Two types of constraints in multi-agents systems can be identified:

- Internal constraints (or intra-agent): whose variables are internal to an agent.
- External constraints (or inter-agent): they are due to executive messages.

To summarize, the set of constraints is the following:

$$C = C_{intra} \cup C_{inter} \tag{5}$$

$$C_{intra} = \{C_1, C_2, ..., C_L\} \tag{6}$$

with L intra-agent constraints, where a given constraint C_i is defined on a set of variables:

$$C_i : \{V_{i1}, V_{i2}, ..., V_{ini}\} \subset V_{Properties} \tag{7}$$

$$C_{inter} = \{C_1, C_2, ..., C_K\} \tag{8}$$

With K inter-agent constraints, where a given constraint C_j is defined on a set of variables :

$$C_j : \{V_{j1}, V_{j2}, ..., V_{jnj}\} \subset V_{CSOP} \tag{9}$$

3.6. The agent's optimization function

By definition an agent pursues some goals. It is driven by a set of tendencies, represented by individual objectives or satisfaction functions to achieve. Therefore the agent is always trying to satisfy its individual objectives while minimizing the harmful interactions (conflicts) and maximizing the positive ones (cooperation) in relation to the other agents and the environment [3].

The agent's behavior is based on the maximization function concerning personal interest and the collective ones via interaction with the other agents. We model this behavior by an optimization function, thanks to the satisfactions model proposed in [7]. Formally, the function to optimize is the sum of three sub-functions to satisfy:

- The satisfaction of the own goals or personal satisfaction, named *P(t)*, expressing personal agent's goals, intentions and engagements.
- The interactive satisfaction *I(t)*, depending on the interaction of the agent with the others (measuring conflicts, cooperation, indifference, etc).
- The acquaintances satisfaction, noted *E(t)* (Empathy), which is computed from other agents satisfaction, representing the altruism of the agents [7].

Therefore, we have for an agent i, at time t, the following instantaneous optimization function:

$$SAT_i(t) = \alpha P_i(t) + \delta I_i(t) + \beta E_i(t) \textit{ with } \alpha + \beta + \delta = 1 \quad (10)$$

Coefficients α, β and δ are time-independent and may vary from an agent to another following the behavior we need. The computation of P, I and E satisfactions depends on the problem to solve. However, keys elements to compute them are given in [7] [2] and are used in the next applicative section.

4. Application to the antennae parameter setting problem

4.1. Overview

We introduce an application based on a real problem, the antennae parameter setting problem. This problem fits into the global process of radiomobile network design [5] which involves three sub-problems: positioning, parameter setting and frequency allocation. We have focused on the parameter setting problem which concerns the maximization of the antenna coverage, the minimization of interferences between antennae and the optimization of the handover (the interference and handover concepts will be defined later). Here the main problem is to determine an optimal adjustment of the antenna emission power enabling the communication with a mobile phone.
The antennae are distributed upon the surface to cover, which is modelled as meshes. For each mesh a propagation model enables to predict the local variation (Fade) of the radioelectric field emitted by every antenna given

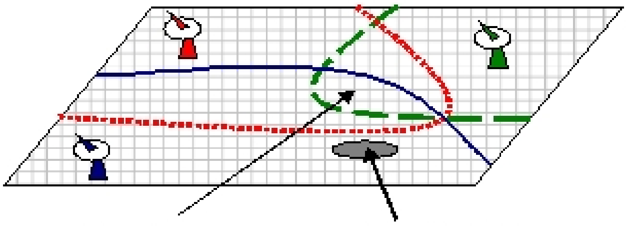

Figure 2. The antennae parameter setting problem

the altitude and the type of ground (see figure 2). To simplify our illustration, we just consider omnidirectional antennae, characterized only by their power parameter.
The antennae parameter setting problem is naturally distributed. Thus we identified two types of agents:

- Agent Antenna: it is characterized by its own variables, e.g. Power, its constraints relating to coverage, its interference and handover rates.
- Agent Environment: we agentify the environment to be able to deal with the problem of not covered meshes more cleverly than from the antennae local point of view. The environment variables are the radioelectric field coming from each antenna (Fade), the number of not-covered meshes (Nb-not-cov) and the number of over-interfered zones (Nb-over-inter). As the agent Environment is parcelled into a set of mesh agents, we define an over-interfered zone as a mesh which has interferences and surrounded by meshes that have also interferences.

4.2. Variables and Values

The agent Antenna and the agent Environment possess respectively the following variables sets:
$V_{Antenna} = \{Power,\ NC,\ E\} = V_{Properties}$
NC is the quantity of traffic elapsed by the antenna a_i (NC_{a_i}) and E is the maximum traffic authorized for the same antenna (E_{a_i}).
$V_{Envir} = \{Nb - not - cov; Nb - over - inter\}$
$\cup \{Fade_{m1};\ Fade_{m2}; ...;\ Fade_{mx}\} = V_{Properties}$
As we break up the environment into a set of meshes (x meshes), the variable Fade is defined for each mesh ($Fade_{m1};\ Fade_{m2}; ...;\ Fade_{mx}$).
The values sets are:
$D_{Power} = D_{initial} \cup D_{interactions}$ where $D_{initial}$ is the value interval given by the operator and $D_{interactions}$

contains values imposed by the agent Environment (see constraints below).
Concerning the agent Environment, the domains of the three variables are included in $D_{initial}$, we don't have values induced from the interaction process, i.e. $D_{interactions} = \emptyset$.

4.3. Constraints

4.3.1. Intra-agent constraints: There is one single intra-agent constraint relating to the agent Antenna. It expresses that the traffic quantity for the antenna a_i must be lower than the maximum traffic authorized for this same antenna (E_{a_i}):

$$NC_{a_i} \leq E_{a_i} \tag{11}$$

4.3.2. Inter-agent constraints

- Coverage constraint: to enable the communication with a mobile phone in a mesh m, i.e. to ensure the mesh coverage, the field received from an antenna a_i must, by definition, be greater than a quality threshold ($G_{quality}$), namely:

$$cov_m^{a_i} : F_m^{a_i} \geq G_{quality} \tag{12}$$

  Where $F_m^{a_i} = Power_{a_i} - Fade_m$.
  $G_{quality}$, as well as $G_{handover}$ and $G_{sensibility}$ in the following constraints, are constants belonging to the agent Environment.
  Note in addition that every mesh must be covered at least by one antenna which implies the following constraint :

$$\forall\ m \text{ (a given mesh) } \exists\ a_i \text{ such that } cov_m^{a_i} \tag{13}$$

- Handover constraint: the notion of handover is used to enable a mobile phone to go from an area covered by one antenna to an area covered by another antenna. A mesh m established a handover relationship between two antennae if it is covered by both and if the difference between the fields received is under the handover threshold ($G_{handover}$):

$$H_m^{a_i,a_j} = \begin{cases} cov\ _m^{a_i} \\ cov\ _m^{a_j} \\ |F\ _m^{a_i} - F_m^{a_j}| \leq G_{handover} \end{cases} \tag{14}$$

- Interference constraint: a mesh m covered by an antenna a_i is interfered by another a_j if the field received from a_j is greater than a sensibility threshold ($G_{sensibility}$):

$$I_m^{a_i,a_j} = \begin{cases} cov\ _m^{a_i} \\ F\ _m^{a_j} \geq G_{sensibility} \\ |F\ _m^{a_i} - F_m^{a_j}| > G_{handover} \end{cases} \tag{15}$$

Our approach allows us to define two other inter-agent constraints, they are defined by two executive messages:
If the number of not-covered meshes is not null, the environment sends an executive message towards the agents Antenna situated in the neighborhood of these meshes. The agents Antenna consider this message as an inter-agent constraint imposing the change of the Power value (increase of Power).
In the same way, if the number of over-interfered zones is greater than a fixed threshold, the agent Environment sends also an executive message towards the close antennae. This message implies also a change of the Power value (decrease of Power).

4.4. Optimization functions

4.4.1. The agent Antenna: it has personal, interactive and altruistic goals. The personal objective consists in the coverage maximization:

$$Maximize\ (NC_{a_i}) \tag{16}$$

$P_{a_i}(t) = NC_{a_i}$ is the personal satisfaction of an agent a_i at time t.
The interactive goal consists of two sub-objectives :

- Minimizing interferences:

$$\forall a_j\ Minimize \sum NI^{a_i a_j} \tag{17}$$

  $NI^{a_i a_j}$ represents the number of meshes covered by the antenna a_i and interfered by the antenna a_j.
- Optimizing the number of handover, i.e minimizing the difference between the optimal number($NH_{optimal}$) and the real number of handover (NH):

$$Minimize\, |NH_{optimal} - NH| \tag{18}$$

The interactive satisfaction which takes into account the interferences and the handover constraints is then defined as follows (I_{max} is a positive constant):

$$I_{a_i}(t) = I_{max} - |NH_{optimal} - NH| - \sum NI^{a_i a_j} \tag{19}$$

The altruistic goal consists on maximizing the acquaintances satisfaction [7], for the agent Antenna it is computed as following:

$$E_{a_i}(t) = \sum_{j \in\ neighbours(a_i)} SATa_j(i \neq j) \tag{20}$$

The optimization function for an agent Antenna a_i is:

$$SAT_{a_i}(t) = \alpha P_{a_i}(t) + \delta I_{a_i}(t) + \beta E_{a_i}(t) \tag{21}$$

These three sub-satisfactions will step-in defining the antenna power variation (see next section). Since the principal goal for an agent Antenna is first the coverage maximization, second the interferences minimization and third the handover optimization, we set $\alpha > \delta > \beta$.

4.4.2. The agent Environment: it has only personal and altruistic goals. The personal goal consists on maximizing the covered meshes, it is expressed as following:

$$P_E(t) = Card\ (m,\ such\ that\ \exists\ a_i\ with\ cov_m^{a_i}) \quad (22)$$

The altruistic goal consists in maximizing the antennae satisfaction:

$$E_E(t) = \sum_{i \in\ antennae\ set} SATa_i \quad (23)$$

To ensure an efficient altruistic behavior the environment sends executive messages (decrease power) only to the agents Antenna affected by the over-interfered zones. These zones are computed by the agent Environment since it has a global vision of the system. Consequently the optimization function of the agent Environment is :

$$SAT_E(t) = \alpha P_E(t) + \beta E_E(t)\ with\ \alpha + \beta\ =\ 1\ and\ \alpha > \beta \quad (24)$$

The coefficient α is dominating, since the environment seeks principally to maximize the number of covered meshes.
Thus, we define the DCSOP global optimization function as the weighted sum of the equations (21) and (24):

$$SAT_{Global} = \alpha SAT_{a_i}(t) + \beta SAT_E(t) \quad (25)$$

Since the global objective of the system is to maximize the coverage, minimize the interferences and optimize the handovers, we assign more importance to antennae satisfaction, justifying $\alpha > \beta$.

4.5. Experiments

In this section we present a distributed solution, based on the previous specification, to optimize the antennae parameter setting. We took use of the Madkit platform to implement our approach. Madkit is a modular and scalable multi-agent platform written in Java and built upon the AGR (Agent/Group/Role) organizational model [4]. This platform allows to show the system evolution and to display curves in real time. It must be emphasized that the data used in experiments correspond to real values belonging to France Telecom.

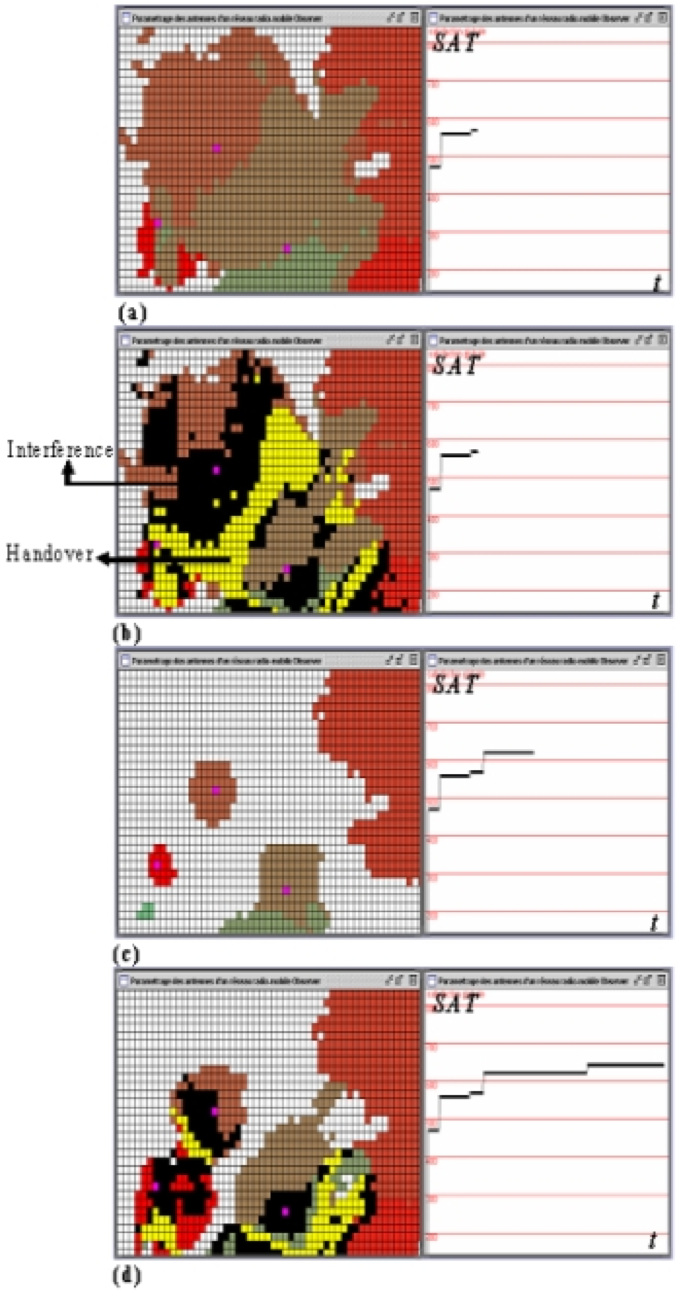

Figure 3. Example of an evolution sequence of the antennae parameter setting problem implementation; Curves represent the global satisfaction evolution; Black areas in (b) represent interferences; Gray-light areas in (b) represent handovers

To evaluate our optimization function specification, we have implemented a reactive *MAS* approach. Agent behaviors are simple and based on the functions defined in section 4.4:

- Antennae behavior: the power changes following the $SAT_{ai}(t)$ (eq. 21). The power value decreases when $SAT_{ai}(t)$ falls, otherwise it increases.
- Environment agent behavior: if $SAT_E(t)$ value goes under a specific threshold, the agent sends to antennae which are close to over-interfered zones an executive message to reduce their power.

Figure 3 shows an example of this implementation. In this example we have 6 antennae and the environment is a grid of 55x63 meshes. Each antenna coverage is represented by a different color. On the left part of the snapshots we visualize coverage, interferences (black areas) and handover (gray light areas).
The system is initialized with huge values for the antennae power. Figure 3.a shows this initial state by drawing only the antennae coverages (interferences and handovers areas are not represented). When the system starts, as an immediate consequence we obtain an explosion of the handovers and the interferences number (see figure 3.b). Forthwith the system reacts and we notice in figure 3.c a decrease of the interferences and the handovers number but at the expense of the coverage. At the same time on the right part of the snapshots, the curves plot the global optimization function (equation 25). Note that the presented curve contains only the best values.
As is shown by the figure 3.d the curve has an asymptotic trend which indicates that the satisfaction has converged and there is no improvement any more (here the convergence is reached in about 800 cycles). This shows that agents search to optimize their functions following an exploration phenomenon.
This first result has been obtained quickly after the problem modeling step. Its shows the relevance of our model and allows to plan the use of *DCSP* algorithms and other distributed solving techniques.

5. Conclusions

In this paper we have provided a generic tool to specify *MAS* with the *CSP* formalism. Generally the contrary is proposed, i.e. using agents to solve *CSP*. That made our approach an original one.
Given a problem and its multi-agent model, we have showed how the *CSP* formalism allows to extract agent's variables, constraints and optimization functions. The knowledge gained at modeling an agent as a *CSOP* and optimization functions is of a singular interest. It determines a formal expression of the *MAS* and allows to solve or to optimize the problem with *DCSP* approaches and tools.
We illustrated our specification approach on the antennae parameter setting problem and we showed the feasibility and the interest of such a model. In particular we have explored a satisfaction-based approach for the specification of optimization functions. Its programming with a multi-agent platform shows a first interesting result in the optimization problem framework. More generally, our approach allows to handle numerous parameters and constraints of a *MAS* in a rigorous way (as shown in the radiomobile network application).
Interesting future directions regard the use of our approach with *CSP* solving algorithms, e.g. Asynchronous Back-Tracking and Maintaining Arc Consistency.

References

[1] R. A. Brooks, "Intelligence without representation", *Artificial Intelligence*, 1991, pp 139-159.

[2] J. Chapelle, O. Simonin, J. Ferber, "How Situated Agents can Learn to Cooperate by Monitoring their Neighbors'Satisfaction", *European Conference on Artificial Intelligence*, 2002, pp 68-72.

[3] J. Ferber, *Multi-Agent Systems: introduction to distributed artificial intelligence*, Addison Wesley, 1999.

[4] J.Ferber, O. Gutknecht, "A meta-model for the analysis and design of organizations in multi-agent systems", *Third International Conference on Multi-Agent Systems*, 1998, pp 128135.

[5] T. Lissajoux, V. Hilaire, A. Koukam, A. Caminada, "Genetic algorithms as prototyping tools for multi-agent systems : application to the antenna parameter setting problem", *Lecture Note in Artificial Intelligence*, 1998.

[6] A.S. Rao, M.P. Georgeff, "Modeling rational agents within a BDI-architecture", *Knowledge Representation and Reasoning*, 1991, pp 473-484.

[7] O. Simonin, J. Ferber, "Modeling Self Satisfaction and Altruism to handle Action Selection and Reactive Cooperation", *the Sixth International Conference on the Simulation of Adaptative Behavior*, 2000, pp 314-323.

[8] M.P. Singh, "A semantic for speech acts", *Annals of Mathematics and Artificial Intelligence*, 1993, pp 47-71.

[9] T. Walsh, "SAT v CSP", *Constraint programming*, 2000, pp 441-456.

[10] M. Wooldridge, N.R. Jennings, D. Kinny, "The Gaia methodology for agent-oriented analysis and design", Autonomous Agents and Multi-Agent Systems, 2000, pp 285-312.