Active mental entities: a new approach to endow multi-agent systems with intelligent behavior

Pietro Baroni*, Daniela Fogli*, Giovanni Guida*, Silvano Mussi°*

* Università di Brescia, Dipartimento di Elettronica per l'Automazione,
Via Branze 38, 25123 Brescia, Italy e-mail: {baroni, guida, fogli}@bsing.ing.unibs.it
° Consorzio Interuniversitario Lombardo per l'Elaborazione Automatica,
Via Sanzio 4, 20090 Segrate (MI), Italy e-mail: mussi@icil64.cilea.it

Abstract

The explicit representation of mental states, such as belief, desire, intention, etc., is a crucial issue for the design of multi-agent systems aimed at performing complex tasks which require some form of global intelligent behavior. In particular, methods for conflict resolution between competing mental processes are a key factor in determining the dynamic behavior of an autonomous agent. In this paper, a novel approach to model agent mental activity based on the concept of active mental entity is proposed. Then, the general organization and operation of a multi-agent architecture encompassing an explicit representation of agent mental activity is introduced. In this context, the issue of conflict resolution at the level of active mental entities is faced and suitable conflict resolution methods are proposed. Finally, an application example concerning a mail delivery robot is presented and discussed.

1. Introduction

The design of intelligent autonomous agents and the development of multi-agent systems has recently received a great deal of attention in the artificial intelligence community. In fact, the realization of distributed systems whose behavior is obtained by the cooperation of intelligent entities is considered as a promising approach to build advanced applications in several fields.

The term *agent* is used in various contexts with different meanings. However, a recent research trend has focused attention on agents conceived as entities "which appear to be the subject of beliefs, desires, etc." [15]. In other words, attitudes such as beliefs, intentions, obligations, desires, etc., are ascribed to agents, in order to obtain an abstract tool "which provides us with a convenient and familiar way of describing, explaining, and predicting the behavior of complex systems" [16]. Among

the theoretical works following this modeling perspective, Cohen and Levesque [4] have proposed a new approach to the problem of formalizing a theory of intention, and Rao and Georgeff [13] have developed a logical framework for agents theory based on the representation of agent mental states in terms of beliefs, desires and intentions (*BDI-theory*). Some of the implementations inspired to these theories are the *Procedural Reasoning System (PRS)* [8] and the *Intelligent Resource-Bounded Machine Architecture (IRMA)* [3] along with its experimental environment *Tileworld* [12]. In both these architectures, mental states are modeled as mere data structures and a central mechanism uses these data structures to make decisions and manage agent operation. An alternative model is proposed by Corrêa and Coelho [5]; in this approach, each agent is endowed with four so-called *local agents* which operate in parallel by managing agent's beliefs, desires, expectations and intentions respectively.

The models of individual agents mentioned above have then been the basis for the development of multi-agent systems. In fact, there exist multi-agent versions both of Tileworld (*MA-Tileworld*) [6] and of *PRS* (the so-called *Oasis* system) [14]. Moreover, in [9] [10] a multi-agent system is proposed (inspired to the BDI-theory and called *Grate**), where the problem of cooperation between agents is managed in an innovative way. In these proposals the issue of coordination and cooperation between agents has been dealt with in different ways. MA-Tileworld uses a filtering strategy that enables each agent to filter out all options incompatible with the objectives of other agents. In Oasis some agents play a specific role in sequencing and coordinating the overall system operation. Finally, Grate* exploits a particular mechanism based on the concept of joint intention which guides the realization of a collective action on behalf of a set of agents. We will discuss this issue in greater detail in section 3.

In this paper, we present an innovative approach to modeling intelligent autonomous agents based on a distributed paradigm. As already mentioned above, an initial step in this direction has been made in [5], where the authors propose to "keep active components in parallel […] inside the whole organization of the agent" in order to keep separate the management of beliefs, desires, expectations and intentions. However, these mental attitudes maintain the role of data structures manipulated by the respective local agents which are persistent inside the "global" agent.

Our basic idea consists instead of endowing mental attitudes themselves with capabilities of autonomous operation. Therefore, we conceive beliefs, intentions, desires, etc. as *active mental entities*, stressing that they can autonomously operate and cooperate. According to our approach, all crucial functions determining agent behavior are performed through the free interaction between active mental entities, which can arise or disappear whenever necessary. In a word, the inside organization

of an agent appears as an ecosystem of mental entities. Therefore, we propose a novel architecture for a multi-agent system which is viewed as a community of agents having an internal dynamic distributed structure.

For the sake of simplicity and brevity we consider here only two classes of active mental entities, namely *intentions* and *persuasions*, which are sufficient in this context to build a significant model of the mental activity of an agent. The overall operation of a multi-agent system is obtained through the interaction and cooperation of such intelligent agents. In particular, we focus here on the mechanisms adopted to solve conflicts occurring between agents, and we claim that they are strictly related to several important aspects of intelligent behavior. We show then that conflict resolution can be carried out at the level of mental entities, and we illustrate how conflicts involving intentions and persuasions can be effectively managed by suitable conflict resolution protocols.

2. A new approach to model agents and multi-agent systems

In this section we illustrate the main concepts of our agent theory (a detailed description can be found in [1] [2]). It is based on the idea of active mental entity, which includes intentions and persuasions. Mental entities constitute fundamental components of individual agents, organized in a multi-agent system.

2.1 Intentions and persuasions

In [4] the authors suggest that "intentions are representations of possible actions the system may take to achieve its goals". According to this point of view, since the role of an action is transforming the current state of the world into a new one, an intention coincides with a desirable state of the world. Differently from this perspective, we claim that an *intention* should be rather conceived as the will of reaching a particular state of the world. In our opinion, an intention should therefore express the concept: "I want to pursue something". Thus, we assume that an intention is an autonomous active entity definitely committed to reach its achievement, also called the *subject* of the intention. To achieve its subject, an intention is able to generate *plans*; that is sequences of *tasks* representing either primitive actions (computations, sensorial acquisitions, actions on the environment) or non primitive actions which become in turn the subject of other intentions. Then, an intention is capable of choosing the most suitable plan and of putting it at work. Finally, it is able to revise the chosen plan, if the external situation changes.

Sometimes, pursuing a particular intention can be impossible or inconvenient or meaningless; for this reason, we assume that intentions must rely on some *validity conditions*. For instance, the intention "I want to find Mr. Smith" is valid only under the condition that it is believed possible to find Mr. Smith and, of course, Mr. Smith

has not been found yet. We distinguish between *generated intentions*, which are created as a consequence of the interaction between other mental entities, and *primitive intentions*, which are always active (such as "I want to preserve my integrity").

In our model, the concept of persuasion substitutes and extends that of belief. "[Beliefs] includes facts about static properties of the application domain […], current observations about the world or conclusions derived by the system from these observations" [8]. In this view, beliefs are conceived as data structures resulting from a perceptual or reasoning activity. We claim, instead, that an agent has a *persuasion* when it is interested in knowing the truth value of a given proposition. For this reason, a persuasion is generated when a new interesting proposition is met and is dismissed when the interest in the proposition ceases. Therefore, a persuasion is a persistent entity, that remains active until the proposition to which it refers is considered interesting. A persuasion is thus not just a passive data structure, but rather an autonomous active entity definitely committed to find and verify elements that can support the belief or disbelief in the truth of a proposition.

The activity of a persuasion is carried out by a process of searching *justifications* about the truth value of the related proposition. The justification for a truth value may be based on long-term knowledge, on sensorial data or, in turn, on other persuasions. Therefore, a persuasion may update or revise the truth value of the associated proposition when long term knowledge changes, new sensory data are acquired or other persuasions have revised the truth value of their own related propositions.

The following basic relationships between intentions and persuasions hold. When an intention is active, its subject and its validity condition become interesting propositions; therefore, relevant persuasions are created. Persuasions are involved also in the phase of plan generation on behalf of an intention. In fact, in general, a particular plan is chosen only if there is the persuasion that it is a viable and effective way for accomplishing the intention. Moreover, during the process of searching evidences to support the validity of the related proposition, a persuasion may require the acquisition of data from the environment; this implies the creation of new intentions carrying out the data acquisition task. In this way, a continuous intention-persuasion interaction takes place. Intentions generate persuasions which may enable or suppress other intentions, which may generate other persuasions, etc. etc.

2.2 Architecture and operation of an agent

An agent features a structured internal micro-organization that includes *components* and *knowledge*. Components are classified into three types:

- *operative components*, in charge of performing actions, either physical, concerning the interaction with external world through sensors and actuators, or symbolic, such as computational and reasoning activities;
- *mental components*, that is intentions and persuasions;
- *interface*, in charge of managing interactions with other agents and the external world.

All agent components are understood as active entities, which can communicate and cooperate in order to produce the global agent behavior. Mental components interact among them and activate operative components which, besides performing actions, may provide a feedback to mental components about the results of the actions carried out. It is assumed that all interactions among components are carried out according to a message-passing paradigm. All the components of an agent share the same basic structure. They are composed of two *modules:*

- the *kernel*, which is in charge of performing the specific tasks the component is capable of;
- the *shell*, which is aimed at supporting communication activities: it is in charge of accepting and filtering incoming messages and of properly addressing messages produced in output.

Turning now to agent knowledge, it represents the basic agent competence endowment, available to all agent components. We distinguish the following basic types of agent knowledge:

- *self knowledge*, that is knowledge concerning agent's own specific features and capabilities;
- *mutual knowledge*, that is knowledge concerning competencies and capabilities of other agents;
- *strategic knowledge*, that is knowledge about the strategies used by intentions and persuasions in their operation;
- *problem-solving knowledge*, that is knowledge concerning the various ways to execute specific tasks;
- *domain knowledge*, that is knowledge concerning the agent competence domain.

Agent operation is the result of the interaction among its internal components. Agent interface is in charge of receiving and processing external problem-solving requests coming directly from the user or from other agents. The interface uses self knowledge in order to check whether the request can be accepted, i.e. if the incoming problem belongs to the agent competence. If this is the case, a way to tackle the problem has to be found out. If the problem is simple and an operative

component able to solve it is available, the problem is directly addressed to it. Otherwise, if the incoming problem is more complex and can not be directly solved by a single operative component, a new intention is generated, whose subject coincides with the solution of the problem. The intention, using strategic and problem-solving knowledge, is in charge of identifying a set of alternative plans for the solution of the problem at hand, of selecting one of them and of putting it to work by resorting to the cooperation of other (mental or operative) components, belonging to the same or to other agents. While the selected plan is executed, each intention is capable, through the cooperation of other mental components, of continuously monitoring the environment and revising the currently active plan if required by changes in the external situation.

2.3 Architecture and operation of a multi-agent system

The proposed distributed architecture is made up of a collection of agents. We assume that each agent is endowed with individual resources and can operate in parallel with the other agents. Moreover, agents can communicate through a message-passing mechanism and can cooperate, according to the benevolence assumption, in order to provide the system with a global intelligent problem-solving behavior. We also assume that a user can ask the system to accomplish one or more tasks; for this reason, a suitable agent inside the architecture is specifically devoted to the interaction with the user. The overall operation of the architecture results from the autonomous operation of the agents; an example of multi-agent system operation is presented in section 5.

3. Where does intelligent behavior come from?

By *intelligent behavior* we mean, in a broad sense, the capability of pursuing complex goals in a rational and justified way, taking into account information about the environment and about the changes occurring in it. In our view, intelligence is related to the way behavior is generated rather than to the external behavior actually showed by a system. A system which behaves in an optimal way in some cases, but completely fails in other situations cannot be considered intelligent; a system that performs correct actions only coincidentally is not really intelligent. An intelligent system is rather a system able to deal with failures, to choose the goals to pursue first and to exploit its limited resources with rationality. Such a system does not succeed or fail in a blind way, but can explain its failures and possibly recover from them. Therefore, it is reasonable to look at the following capabilities as particularly important and useful, though not sufficient (in particular we do not consider here learning capabilities), in order to characterize an intelligent behavior:

1. the capability of pursuing multiple goals, managing several different plans in parallel;
2. the capability to cope with unforeseen changes in the environment by revising current plans;
3. the capability to face uncertain situations.

In order to rationally choose between alternative plans (first capability), the agent should have an explicit representation of the motivations that lie behind its behavior. It should be able to reason about the intentions from which its behavior derives, in order to tailor the behavior to meet such intentions in the most appropriate way, coherently with the current state of the environment (second capability). Moreover, the system should be able to make decisions by considering the uncertainty and ambiguity affecting real-world situations (third capability).

In recent approaches to the design of multi-agent systems such capabilities are obtained basically in the same way, namely through some arbitration mechanism among different autonomous entities, which are competing for the allocation of system's resources. In particular:

- The coordination model of MA-Tileworld [6] is based on the so-called *multi-agent filtering*. Different filtering techniques has been adopted and implemented to realize it. The most general and efficient one is *intention posting*: agents post their intentions into a shared structure and filter out all the intentions already declared by other agents. The resulting arbitration mechanism is very simplified and limited; in fact, it has been experimented in the Tileworld environment, but it seems not sophisticated enough to be extended to more realistic contexts.
- The multi-agent system Oasis [14] is endowed with specific agents (namely *global agents*) which play the role of sequencers and coordinators. Therefore, the whole arbitration is managed in a centralized manner through the global agents. This approach lacks flexibility and extendibility since global agents must be modified every time new agents or a new functionality are added to the architecture.
- The arbitration mechanism exploited by Grate* [9] [10] is called *joint responsibility* and is based on the notion of joint intention mentioned in section 1. Quoting Jennings: "joint intentions can be intuitively defined as a joint commitment to perform a collective action". In fact, common goals of the system can be reached by establishing, through a two-phase negotiation protocol, a *social action*. To carry out this common activity, an agent exists which acts as an organizer by controlling the deployment of the social action. If new objectives arise during the pursuing of an intention, a conflict occurs. It is solved by a mechanism, the *inconsistency resolver*, which refers to the agents' desires: "if the new task is less important (desirable) than existing ones, then it is the one which

should be modified; conversely if it is more desirable then it is the existing one which should be adapted" [9]. In other words, a predefined priority order between desires enables agents to determine the action to be performed first.

All the architectures mentioned above exploit a centralized mechanism to solve conflicts: in MA-Tileworld a shared structure is used to determine the intentions to pursue first, in Oasis global agents decide the sequencing of actions, and, finally, in Grate* the inconsistency resolver is a centralized mechanism which analyses the inconsistencies and resolves them by modifying intentions and by deciding their scheduling. In our view, since conflict resolution is the most critical aspect in a distributed control architecture, it should be distributed as well, in the sense that agents should be able to manage their conflicts autonomously, through a proper conflict resolution protocol. The following section is entirely dedicated to this important topic.

4. Conflict resolution as a cooperation strategy

A significant part of the cognitive activity of our system is constituted by interaction and conflict resolution between mental entities. In fact, conflicts between mental components occur every time a situation must be disambiguated, and conflict resolution enables the system to cope with real-word uncertainty in a sophisticated manner.

In case of conflict between intentions, conflict resolution allows the system to deal with multiple, possibly contrasting, goals, and in case a conflict occurs between persuasions, it provides a way to cope with the uncertainty that affects the perception of the world. We will discuss these two different types of conflict resolution in the following of this section, whereas we will present some application examples in section 5.

4.1 Conflict resolution between intentions

A conflict between two intentions may arise when they try to access to a shared resource. If the intentions involved are primitive ones, we assume that a *priority* attribute makes it possible to directly establish the prevailing intention. If one conflicting intention is primitive and the opponent one is not, the latter refers to the primitive intention underlying it, to which the conflict resolution is delegated. Finally, if both intentions are not primitive, conflict resolution involves a more articulated interaction protocol.

Even though other methods could be envisaged, for the sake of brevity, we consider here only a simple protocol, where the execution of the plan of a conflicting intention has to be postponed to that one of the opponent. In order to decide which plan should be postponed, first of all each intention simulates the execution of a new

compound plan, which is obtained from the previous one by including the accomplishment of the subject of the other intention. In order to perform the simulation, the following attributes of the intentions and plans involved in the conflict are exploited:

- a *deadline* attribute associated to intentions, which states a time limit for the accomplishment of the intention;
- a *time-to-finish* attribute associated to the plans used by intentions, which states the estimated time necessary to complete the execution of the plan.

Each intention estimates the *time-to-finish* of the compound plan, by deriving it from the *time-to-finish* values of the individual plans currently used by the intentions involved in the conflict. Then, each intention compares its *deadline* with the *time-to-finish* value of the simulated plan by checking if they are compatible, i.e. if the *time-to-finish* is not greater than the *deadline*. In other words, each intention verifies whether its deadline can be respected even in case of postponement. At this point, three situations can occur:

(i) both intentions recognize that, in their simulated plan, *deadline* and *time-to-finish* are compatible (that is both intentions can accept to be postponed): then, they exchange the *time-to-finish* values of simulated plans in order to select that one which guarantees the faster execution;

(ii) only one intention is compatible with the simulated plan: this means that the accomplishment of this intention can be postponed until the other one has been achieved;

(iii) both intentions are incompatible with respective simulated plans, that is both are unable to meet their deadline in case of postponement; in this case, it has to be decided which of the intention will not be accomplished within the deadline. This decision can not be taken by the conflicting intentions themselves, since it should be related to the deeper motivations underlying the adoption of each intention. Therefore the conflict is transferred at the level of the intentions which generated the conflicting ones, where the process of conflict resolution restarts. The conflict may eventually reach the level of primitive intentions, where it can be directly solved.

4.2 Conflict resolution between persuasions

A conflict between persuasions arises, and needs to be solved, in the two following cases:

- when a persuasion supporting an intention (or a plan generated by an intention) relies on contradictory persuasions, that is on persuasions which ascribe a different truth value to the same proposition;

- when an intention, while elaborating a strategy of action, detects a conflict between two persuasions that can influence the selection of the plan to be adopted.

The conflict resolution protocol for persuasions includes two main phases. The first phase implements a simple and efficient way of solving a conflict through a comparison between the justification types (see below) of persuasions. The second phase, which starts if the previous one fails, is carried out through a debate between the persuasions involved in the conflict. Let us analyze these mechanisms in some detail.

Phase 1. The currently believed truth value of a persuasion is in general justified by a chain of propositions. At the root of this chain, a terminal node represents a justification which can be of one of the following types:

- *default*, for propositions that are usually considered true, but can be false in certain situations;
- *stored data*, for propositions representing information assumed as definitely known;
- *acquired data*, for propositions related to data directly acquired from the external world.

We hypothesize that there exists a strength order between the above justification types: default is less strong than stored data, which, in turn, is less strong than acquired data. In fact, it is reasonable to believe that stored and acquired data are more reliable than default data and, moreover, that acquired data can be considered more reliable than stored data, since the former are more up-to-date than the latter. During the first phase of conflict resolution, persuasions compare their justification types and their relative strength: if they are different, the persuasion having the strongest justification prevails and the conflict is thus solved.

Phase 2. If both persuasions have the same justification type, a more detailed analysis of the actual justifications supporting conflicting persuasions is necessary to solve the conflict. Each persuasion requests more detailed information about the opponent's justifications; then, if such justifications present some weak points, every persuasion chooses one of these and attacks the opponent about it. Then, it waits for an answer. When a persuasion receives an attack, it actuates a defense strategy. Defense is carried out by looking for stronger justifications about the attacked weak point: if new justifications are found, these are communicated to the opponent persuasion. In case a persuasion is not capable of making an attack or of answering to an attack, it notifies its renunciation. When a persuasion receives a renunciation

message, it prevails and its truth value is considered valid. Finally, if both persuasions renounce (or if neither one renounces), the conflict is not solved at this level. This may require more complex resolution methods, involving the simulation of the consequences of the alternative choices or some specific activity aimed at learning new knowledge that may allow the solution of the conflict. However, the discussion about these methods is beyond the limits of this paper.

5. An application example

In this section, we present an example in order to support a better understanding of the organization and operation of our architecture. In particular, we will examine two situations in which conflicts (one between intentions and another between persuasions) arise. The example concerns a department mail delivery robot [1], to which the user consigns an envelope to be delivered to Mr. X. The robot is conceived as a multi-agent system where each agent is able to perform a specific task such as managing a sonar sensor, managing a TV camera, controlling movement actuators, etc.

Let us suppose that the primitive intention whose subject is "obey-the-user" is active into the UI (User Interaction) agent of the robot. After receiving the request of delivering mail to Mr. X, a new intention has to be generated whose subject is "deliver-mail-to-Mr.X". However, since UI has no specific competence on mail delivering, it has to address the request of creating this new intention to another competent agent. By resorting to its interface, UI identifies the MD (Mail Delivery) agent and forwards the request to it. The new intention "deliver-mail-to-Mr.X" is therefore created within MD. This intention may then generate different plans for its achievement. For instance, a simple plan relying on the persuasion "Mr. X is in his office" may be:

Task 1: go to the office of Mr. X;

Task 2: deliver the envelope to Mr. X.

Task 1 is considered first: it still concerns a quite generic and high-level task and must therefore be associated to a new intention. A request of generating such an intention is therefore addressed by MD to MM (Movement Manager) agent. While the intention "go-to-the-office-of-Mr. X" is trying to accomplish itself, a lot of different situations can occur outside and inside the robot. We will analyze below two significant cases.

5.1 Solving a conflict between intentions

Let us assume that, during the movement toward the office of Mr. X, the robot energy reaches the minimum threshold: inside the EC (Energy Control) agent, an

active persuasion whose subject is the "battery is drying up" and which is monitoring the battery situation, notices this fact and modifies its truth value. This change enables the applicability condition of a new intention whose subject is "recharge-battery" which has the following associated plan:

Task 1: go to the recharging point

Task 2: wait for the complete battery recharge.

Task 1 leads to the generation of the new intention "go-to-the-recharging-point" in the MM agent, which immediately starts its activity by carrying out a proper movement plan. Now, both intentions "go-to-the-office-of-Mr.X" and "go-to-the-recharging-point" need to resort to the movement actuators and, therefore, they may conflict one another. Let us suppose, for example, that the recharging point and the office of Mr. X are in opposite directions. Then, at a certain time, intentions "go-to-the-office-of-Mr.X" and "go-to-the-recharging-point" could have to perform the actions "turn-left" and "turn-right" respectively. Obviously, a conflict arises. In order to solve the conflict, intentions may refer to the *deadline* attribute associated to them or, if they do not have it, they may delegate the conflict to their generating intentions. We assume that the *deadline* attribute is assigned from outside; it can be present inside user commands or into domain knowledge. For some intention, a deadline can not be significant or necessary; in this case the deadline will have an unknown value. In our example, it is reasonable to assume that intentions "go-to-the-office-of-Mr.X" and "go-to-the-recharging-point" have an unknown deadline; for this reason, the conflict resolution is carried out first by their generating intentions, such as "deliver-mail-to-Mr. X" and "recharge-battery" respectively. To decide the right scheduling, intentions perform a plan simulation:

- "deliver-mail-to-Mr.X" estimates the time-to-finish of a new plan where mail delivery is postponed to the battery recharging activity;
- "recharge-battery" estimates the time-to-finish of a plan which delays the recharging activity after mail delivery.

These estimates are derived from the values of time-to-finish associated to current plans. Let us suppose now that both simulated plans are incompatible with the deadline of conflicting intentions. In this case, conflict resolution cannot be carried out at the level of "deliver-mail-to-Mr.X" and "recharge-battery". Therefore, again, the conflict is delegated to the generating intentions, which are the primitive intentions "obey-the-user" and "take-care-of-your-safety". Each of these intentions has a priority attribute which allows one to discriminate unequivocally between them. In this case, we can reasonably suppose that "take-care-of-your-safety" has more priority than "obey-the-user", so it wins the conflict and propagates this information to the

other intention involved in the conflict, whose accomplishment will be necessarily postponed.

5.2 Solving conflict between persuasions

Let us now restart from the situation in which "go-to-the-office-of-Mr. X" was the only intention active inside MM. As already stressed, intentions are continuously looking for better plans and persuasions are continuously looking for new evidences supporting them. So, while the robot is moving towards the office of Mr. X, the intention "deliver-mail-to-Mr. X" of MD may elaborate the following alternative plan, relying on the persuasion "Mr. X is not in his office":

Task 1: find Mr. X around in the department

Task 2: go near Mr. X

Task 3: deliver the envelope to Mr. X.

The persuasion "Mr. X is not in his office" is then activated and is in charge of finding support. For this reason, it requires the generation of intentions "recognize-voice-of-Mr.X" and "recognize-face-of-Mr.X" to SRS (Sonar Range Sensors) agent and VC (Video Camera) agent respectively, so creating a sort of mechanism of attention to the presence of Mr. X in the neighbourhood.

Let us suppose now that the robot is near a glass wall behind which there is Mr. X. Thanks to the activity of intention "recognize-face-of-Mr.X", the vision system recognizes Mr. X in front of the robot and, therefore, persuasion "Mr. X is not in his office" gets stronger support. At the same time, whereas "Mr. X is in his office", which is justified by default knowledge that an employee is normally in his office, is discarded. In fact, in this case the first step of conflict resolution protocol enables the immediate solution of the conflict. Accordingly, the plan relying on "Mr.X is not in his office" is preferred and, since Task 1 has been achieved (find Mr. X around in the department), Task 2 is pursued (go near Mr. X). The intention "go-near-Mr. X" is created inside MM and, since Mr. X is standing just in front of the robot, the intention of navigating towards such a fixed target is reduced to the atomic operation "go-forward".

While the robot is moving forward, VC and SRS acquire and process data about the external world. In doing this, they continuously generate or update persuasions about the environment, whose subject is communicated to the intention "avoid-collision" active within the agent IP (Integrity Preservation).

Suppose now that, while the robot is approaching the target, VC and SRS communicate to "avoid-collision" two contradicting persuasions: SRS supports the persuasion "there is an obstacle on the path", whilst VC supports the persuasion "no obstacle on the path". The intention recognizes that it is impossible to take a decision,

given these contradicting persuasions, and therefore decides that the conflict should be solved. It puts the two persuasions face to face by notifying each of them of the existence of the opponent persuasion.

Persuasions "there is an obstacle on the path" and "no obstacle on the path" enter therefore a debate in order to solve the conflict. First of all, an analysis of the justifications supporting them is carried out: "there is an obstacle on the path" is supported by the fact that sonar received reflected echoes, "no obstacle on the path" is supported by the fact that in the image acquired by video camera nothing but Mr. X is seen. Since they are both supported by acquired data, persuasions "there is an obstacle on the path" and "no obstacle on the path" must look for evidence corroborating their justifications or undermining the opponent's ones. For instance, "no obstacle on the path" can resort to general knowledge that sonar readings are often erroneous and attack "there is an obstacle on the path" about this weak point. In turn "there is an obstacle on the path" may reply that sonar readings are erroneous in specific conditions (near wall corners, in presence of noise sources, etc.) that are not met in the present case. Moreover, "there is an obstacle on the path" may attack directly "no obstacle on the path" support resorting to general knowledge, provided by BT (Building Topology) agent, that, in the building, there are invisible obstacles (such as transparent glass walls). Since "no obstacle on the path" is not able to reply to these arguments, "there is an obstacle on the path" prevails: the presence of an obstacle is accepted and "avoid-collision" intervenes to modify the motion plan, going around the glass wall and eventually reaching Mr. X.

6. Discussion and conclusions

The proposal we have presented in this paper relies on three fundamental claims:

1. an agent (and consequently a multi-agent) architecture should include an explicit representation of mental activity;
2. several important aspects of intelligent behavior are related to the way conflicts between agents competing for system control are solved;
3. conflict resolution should be carried out at the level of mental entities, using suitable conflict resolution protocols.

Claim 1. has been raised and has received consensus, in recent years, both from the theoretical point of view [4] [13] and from the practical point of view [8] [12] [5] [9] [10] [6] [14].

Claim 2. though not often made explicit, is common to various recent architectural approaches. In fact, as already mentioned in section 3, the multi-agent filtering in MA-Tileworld or the joint responsibility mechanisms in Grate* are crucial for the effectiveness of the respective architectures: the overall coordination between

agents entirely relies on them. A similar role is played by the global agents in the Oasis system. We believe that explicitly recognizing the role of conflict resolution is very important, since this aspect deserves a greater level of attention than it has received in the past. This work is intended to be a contribution in this direction.

Claim 3. is a direct consequence of claims 1. and 2. Given the crucial role played by conflict resolution between agents in determining system intelligence, it should be related with mental activity, in the sense that conflicts should be solved on a rational basis, taking into account current mental attitudes of the agents composing the system. Though this idea is practically and implicitly included within other existing approaches, our proposal represents a significant step further in the direction of a sophisticated modeling of conflict resolution.

The importance of managing contradiction in practical reasoning has been recently remarked in [7], which examines conflicts in the context of argumentation systems. This work however mainly deals with representation of conflict situations, without entering the issue of conflict resolution. A conflict resolution mechanism based on argumentation is presented in [11]. Similarly to our approach, it encompasses a representation of the mental attitudes involved in the conflict and is based on a negotiation protocol. Their conflict resolution mechanism is however based on a criterion which takes into account a logical classification of conflicting arguments, whereas our methods are intended to take into account practical aspects related to the application context, such as the different importance of primitive intentions or the reliability ascribed to different information sources.

Being based on the claims listed above, our approach represents, as to our knowledge, an innovative point of view both in modeling agent mental activity and in designing multi-agent systems. A software implementation of the proposed architecture is in progress: it will allow a systematic experimentation on a set of complex test cases, in order to better identify merits and weak points of the proposed architecture.

References

[1] P. Baroni, D. Fogli, G. Guida, S. Mussi, An Advanced Control Architecture for Autonomous Mobile Robots: Modeling Intentions and Persuasions. *Proc. of the 4th Int. Conf. on Control, Automation, Robotics and Vision*, Singapore, 1996, 853-857.

[2] P. Baroni, D. Fogli, G. Guida, S. Mussi, Active Mental Entities: a new approach to building Intelligent Autonomous Agents, *Tech. rep. 199702-10, University of Brescia*. To appear in *ACM Sigart Bulletin*.

[3] M. E. Bratman, D. J. Israel, M. E. Pollack, Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4, 1988, 349-355.

[4] P. R. Cohen, H. J. Levesque, Intention is choice with commitment. *Artificial Intelligence*, 42(3), 1990, 213-261.

[5] M. Corrêa, H. Coelho, Around the Architectural Agent Approach to Model Conversations. *Proc. of the 5th European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, 1993, 172-185.
[6] E. Ephrati, M. E. Pollack, S. Ur, Deriving Multi-Agent Coordination through Filtering Strategies. *Proc. of the 14th Int. Joint Conf. on Artificial Intelligence*, 1995, 679-685.
[7] J. Fox, P. Krause and S. Ambler, Argument contradictions and practical reasoning, *Proc. of ECAI 92 10th European Conf. on Artificial Intelligence*, 1992, 623-627.
[8] M. P. Georgeff, A. L. Lansky: Reactive Reasoning and Planning. *Proceeding of the 6th Nat. Conf. on Artificial Intelligence (AAAI-87)*, Seattle, 1987, 268-272.
[9] N. R. Jennings, Specification and implementation of a belief-desire-joint-intention architecture for collaborative problem solving. *International Journal of Intelligent and Cooperative Information Systems*. 2(3), 1993, 289-318.
[10] N. R. Jennings, Controlling cooperative problem solving in industrial multi-agent systems using joint intentions. *Artificial Intelligence*, 75(2):195-240, 1995.
[11] S.Parsons and N.R. Jennings, Negotiation through argumentation-a preliminary report, *Proc. of ICMAS 96, 2nd Int. Conf. on Multi Agent Systems*, Kyoto, 1996.
[12] M. E. Pollack, M. Ringuette, Introducing the Tileword: Experimentally evaluating agent architecture. *Proceedings of the Eighth National Conference on Artificial Intelligence (AAAI-90)*, Boston, MA, 1990, 183-189.
[13] A. S. Rao, M. P. Georgeff, Modeling Rational Agents within a BDI-Architecture, *Proc. of KR&R-91 Int. Conf. on Knowledge Representation and Reasoning*. Cambridge, MA, 1991, 473-484.
[14] A. S. Rao, M. P. Georgeff, BDI Agents: From Theory to Practice, *Proc. of First Int. Conf. on Multi-Agent Systems*. San Francisco, CA, 1995, 312-319.
[15] N. Seel, *Agent Theories and Architectures*. PhD thesis, Surrey University, Guildford, UK, 1989.
[16] M. Wooldridge, N. R. Jennings, Agent Theories, Architectures, and Languages: A Survey, in M. J. Wooldridge, N. R. Jennings (eds.), *Intelligent Agents*, Springer-Verlag, 1995, 1-39.