

Role-Assignment in Open Agent Societies

Mehdi Dastani Virginia Dignum Frank Dignum
Institute of Information and Computing Sciences
Utrecht University
The Netherlands
{ mehdi , virginia , dignum }@cs.uu.nl

ABSTRACT

Open systems are characterized by heterogeneous participants which can enter or leave the system at will. Typical examples are e-commerce applications or information agent systems. Agents (e.g. personal assistants for buying things on the Internet) will only temporarily take up roles (e.g. a buyer in on-line auctions). This creates the need to define precisely what it means that an agent "takes up" a role and "enacts" it. In this paper we present ongoing research on the determination of the conditions under which an agent can enact a role and what it means for the agent to enact a role. We define possible relations between roles and agents and discuss architectural and functional changes that an agent must undergo when it enters an open agent system.

Categories and Subject Descriptors

I.2.11 [Artificial Intelligence]: Distributed Artificial Intelligence—*multi-agent systems*; D.2.10 [Software Engineering]: Design—*methodologies*

General Terms

Design, Languages, Theory

Keywords

Role, Organization, Norm, Goal, Conflict, Enacting Role

1. INTRODUCTION

In the emerging model of electronic business, a variety of open electronic societies and marketplaces will be competing with one another for participants. Independently developed agents enter and leave marketplaces at will, in pretty much the same way that human investors enter and leave different financial markets today [4]. This situation illustrates the need for systems that are open to the participation of different, heterogeneous agents. Open systems are characterized by heterogeneity of participants, limited trust, conflicting

individual goals, and a high probability of non-conformance to specifications [1]. Multi-Agent Systems (MAS) is one of the most promising approaches to create open systems because of their ability to dynamically reorganize themselves as their problems and constituent agents change [10]. Nevertheless, most existing MAS architectures are closed, in the sense that participating agents must be designed following a given internal architecture. In our opinion, the effective design of open agent societies requires the following aspects to be considered. First, formal frameworks are needed that specify the society structure and goals with verifiable and meaningful semantics in a way that is independent from the participating agents. Second, we need to specify mechanisms through which prospective participants can evaluate the characteristics and objectives of society roles, in order to decide about participation. Third, the tools for individual agents to adapt their architecture and functionality to the requirements of an assumed role must be provided.

The importance of the first point has been widely acknowledged. Society structure is often determined by organizational design which is independent of the agents themselves. Agent Societies emerge from the idea that agent interactions occur not just by accident but aim at achieving some desired global goals. That is, there are goals external to each individual agent that must be reached by the interaction of those agents. Desired behavior of a society is therefore external to the agents. Currently, several formalisms have been presented concerning the first point above [1, 5, 6]. Such formalisms describe the intended organizational structure of the society and the laws that govern interaction among agents in the same way as institutions are used in human societies to lend legitimacy and security to its members by establishing norms. With respect to the second and third points, considerable work is needed and many research areas are open.

The second point refers to the need to match agent and role objectives and functionality. At the moment most approaches to this problem simply design agents from scratch so that its behavior complies with the behavior described by the role(s) it will take up in the society. This applies especially to closed MAS where all agents are designed centrally and with a pre-intended purpose. In this case the design of the agent follows from the requirements specified for the role(s) the agent is fulfilling. See e.g. Gaia [13] for a good example of this perspective. Comprehensive solutions for this problem require complex agents that are able to reason about their own objectives and desires and thus decide and negotiate their participation in a society. A first step on

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

AAMAS'03, July 14–18, 2003, Melbourne, Australia.
Copyright 2003 ACM 1-58113-683-8/03/0007 ...\$5.00.

the road to this solution (taken in this paper) is to have a formalism to compare the specifications of agents and roles and determine whether an agent can enact a role. In the future, agents themselves will be able to use this mechanism to automatically evaluate their participation in a society.

Finally, the third point above indicates that, once a decision has been reached that an agent will indeed enact a role, there must be ways to modify that agent in order to include the characteristics of the assumed role. A possible solution for this point has been proposed in [12] in which agents are extended with an interface to the society. This interface prevents any action not allowed by the role definition. However, it does not facilitate proactive behavior expected from the agents while playing the role. We think that the consequence of an agent adopting a role is more drastic than this. The actual agent behavior must be changed according to the objectives, norms and capabilities specified by the role. In [2] it is assumed that agents have to adopt the goal (or responsibility) of the role once they take up that role. Our proposal allows for different approaches to such modification of an agent which result in different role performances. For instance, some agents will only attempt to achieve the goals of its adopted role and forget its own private goals, while others will only attempt to achieve the goals from the role after its own goals have been satisfied. The consequences of taking up a role are also discussed in [7]. Here the roles are characterized by a set of norms that the agent should obey when playing that role. Although we also assume the role to contain a set of norms, we also associate a set of goals with the role. The main difference is that a goal (stemming from the role that the agent plays) represents an intrinsic motivation of the agent enacting the role to pursue a course of actions, while a norm is an extrinsic motivation [3].

In this paper we describe initial research concerning the consequences of role-assignment using a fixed society description as presented in section 2. In section 3 we present a formal abstract definition of agents and roles based on the formal agent programming language 3APL. Section 4 is about agents enacting roles. This section starts by defining coherent agents and coherent roles, and goes on describing how an agent can be modified to allow for the enactment of a role. In section 5, we discuss open issues and future research. Finally, in section 6 we present our conclusions.

2. AGENT SOCIETIES

In [5] a model for agent societies is introduced that takes an organizational perspective. This model describes an agent society in three levels.

1. **Organizational model (OM)**: it describes the desired or intended behavior and overall structure of the society from the perspective of the organization in terms of roles, interaction scripts and social norms.
2. **Social model (SM)**: it populates the organizational model with agents that are mapped to roles through a social contract. Social contracts describe the agreed behavior for an agent within the society in terms of externally observable events. Agents enacting a role are called actors.
3. **Interaction model (IM)**: it specifies interaction agreements between actors and describes the actual behavior of the society.

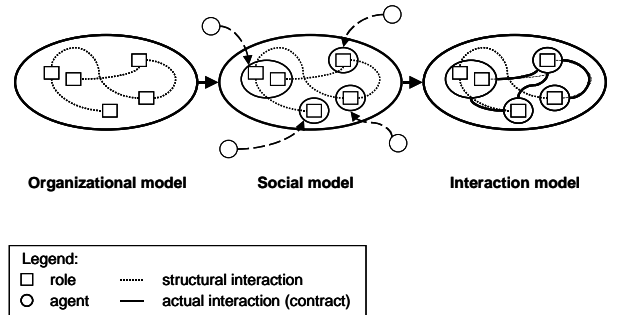


Figure 1: *Agent society model*

The relation between these models is depicted in figure 1. For the purpose of this paper, we will be referring only to the OM. The OM can be more or less elaborate, depending on the type of society. It must describe the different roles participating in the society and the way interaction between roles is expected to happen in order for the society goals to be realized. In general, roles interact with each other in different interaction scenes. For the sake of simplicity, in this paper we assume that an organizational model consists of only one interaction scene. That is, the society structure is completely defined by a set of roles. Typically, interaction is described using "landmarks" that specify the main features (conditions, obligations, and possibly partial plans) of interactions between roles. It is left to the agents how to achieve these landmarks. For a complete specification of the society model we refer the reader to [5]. In the rest of this section, we will informally describe the concepts of role and agent as used in the Agent Society Model and introduce the working example which will be used in the rest of the paper.

2.1 Organizational roles

Roles are one of the basic components of an organizational model. A role is the abstract representation of a policy, service or function. Role descriptions in the organizational model identify activities and services necessary to achieve society goals and enable us to abstract from the individuals that eventually will enact the role. Furthermore, roles must describe the necessary capabilities that must be performed by any agent enacting that role. Roles are organized into a role-relationship network.

In our model, roles can be seen as place-holders for agents and represent the behavior expected from agents by the society design. Roles are specified by goals, norms, and interaction rules. The goals of a role describe the results that the role enacting agent must seek to obtain. They are equal to what in other approaches such as [13] are called the responsibilities of the role. Norms of a role specify the obligations and prohibitions of the role enacting agents. Interaction rules describe the interaction between roles and specify how role enacting agents should interact with each other. A formal description of a role will be given in section 3.3.

2.2 Agents

In our model, agents are active entities that are able to enact roles described in the Organizational Model of a society. Agents join a society by adopting some of its roles. In order to be able to take up a role in the society, it is necessary for an agent to alter (extend, modify or limit) its own behavior

such that it will react within the society in ways that are in accordance to the expectations of the role.

We assume that agents are designed outside the scope of the society, and have their own goals, beliefs, capabilities, and reasoning rules. An agent will seek admittance to the society at a given moment of its life cycle (the reasons for which we do not consider here). We have kept our formalism as close to current agent implementations as possible. Because most existing agent architectures do not provide (yet) the possibility to specify normative agents we choose not to include norms in our formalism for the agent (but only include them in the role definition). A formal description of an agent will be given in section 3.2.

2.3 Working Example

In the remaining of this paper, we will illustrate role related concepts using an example of conference organization. The process of organizing a conference can be modelled as an agent society whose global goal is the realization of a conference. This goal is achieved through the interaction of several roles which are specified in the Organizational Model. We assume that this society contains the roles of *Organizer*, *Submitter*, *PC – member*, *Author*, *Presenter* and *Participant*. The goal of *Organizer* is to realize a successful event for which it will need to rely on the *PC-members* whose goal is to review the papers submitted. The *Author* goal is to get its submission accepted, in which case an author becomes a *Presenter* during a conference session. The *Participant* goal is to attend a high quality conference.

3. AGENTS AND ROLES

In the following we present an abstract formal language to specify cognitive agents and roles. We assume that a translation between any cognitive agent description language and this formalism is possible. For the sake of illustration we have based this formalism on the 3APL language [8].

3.1 3APL: Syntax and Semantics

In this section, we assume an agent model which we believe to be adequate for enacting roles in agent organizations. Such agents, also called 3APL agents, have beliefs and goals. In order to achieve their goals 3APL agents reason about their beliefs and goals, and execute actions. We believe that this agent model is adequate for three reasons. The first reason is that 3APL agents have the necessary ingredients (concepts and tools) to specify a role that can be assigned to them. The second reason is that 3APL agents have a formal syntax and semantics such that they can be specified unambiguously. The third reason is that the proposed formal syntax and semantics is computational such that agent organizations (i.e. a set of agents each of which plays a role) can be implemented. Since the formal syntax and semantics of 3APL is already introduced in [8], we only present a brief summary in order to make this paper self contained.

3APL agents consist of four modules. These modules are beliefs, basic actions, goals and reasoning rules. Each module can be specified by means of expressions of a language. Below is an overview of the languages for each module. The first language is the belief language the expressions of which specify the agent’s information about its environment.

DEFINITION 1. Given a set of domain variables and functions, the set of domain terms T_D is defined as usual. Let $t_1, \dots, t_n \in T_D$, $Pred_b$ be the set of belief predicates, and $p \in Pred_b$. The belief language \mathcal{L}_B is defined as:

$$\mathcal{L}_B = \phi ::= p(t_1, \dots, t_n) \mid \neg\phi \mid \phi \wedge \psi$$

All variables in $\phi \in \mathcal{L}_B$ are universally quantified with maximum scope. The beliefbase module of a 3APL program is a set of belief formulae.

The set of basic actions is a set of (parameterized) actions that can be executed if certain preconditions hold. After execution of an action certain postconditions must hold. These actions can be, for example, physical actions or belief update operations.

DEFINITION 2. Let $t_1, \dots, t_n \in T_D$, $Action$ be a set of action names, $A \in Action$, and $\phi, \psi \in \mathcal{L}_B$. Then, the action language \mathcal{L}_A is defined as:

$$\mathcal{L}_A = \alpha ::= \langle \phi, A(t_1, \dots, t_n), \psi \rangle$$

The belief formula ϕ is called the pre-condition of the action and must hold before performing the action. The belief formula ψ is called the post-condition of the action and must hold after performing the action. The basic action module of a 3APL program is a set of basic actions.

The set of goals consists of different types of goals: Basic action goals (BactionGoal), predicate goal (PredGoal), Test goal (TestGoal), skip goal (SkipGoal), sequence goal (SeqGoal), if-then-else goal (IfGoal), and while-do goal (WhileGoal).

DEFINITION 3. Let $t_1, \dots, t_n \in T_D$, $\alpha \in \mathcal{L}_A$, $\phi \in \mathcal{L}_B$, $Pred_g$ be the set of goal predicates such that $Pred_b \cap Pred_g = \emptyset$ and $q \in Pred_g$. Then, the goal language \mathcal{L}_G is defined as follows:

$$\mathcal{L}_G = \pi ::= \alpha \mid q(t_1, \dots, t_n) \mid \phi? \mid skip \mid \pi_1 ; \dots ; \pi_n \mid \text{IF } \phi \text{ THEN } \pi_1 \text{ ELSE } \pi_2 \mid \text{WHILE } \phi \text{ DO } \pi$$

The goal base module of a 3APL program is a set of goals.

It is important to note that the notion of goal here is a kind of to-do-goal which is closely related to the notion of task or plan. Moreover, predicate goals, which play an essential part in this paper, should be seen as task names. Predicate goals should not be confused with belief formulas and therefore should not be considered as denoting a state. For example, unlike a belief formula, a predicate goal cannot be composed with other predicate goals by means of logical operators to form more complex predicate goals.

In order to define practical reasoning rules, a set of goal variables, $GVAR$, is introduced. These variables are different from the domain variables used in the belief language. The goal variables may occur in the head and the body of practical reasoning rules and will be instantiated with a goal. Note that domain variables are instantiated with the belief terms. We extend the language \mathcal{L}_G with goal variables. The resulting language \mathcal{L}_{G_v} extends \mathcal{L}_G with the following clause: if $X \in GVAR$, then $X \in \mathcal{L}_{G_v}$.

DEFINITION 4. Let $\pi_h, \pi_b \in \mathcal{L}_{G_v}$ and $\phi \in \mathcal{L}_B$, then a practical reasoning rule is defined as: $\pi_h \leftarrow \phi \mid \pi_b$.

This practical reasoning rule can be read as follows: if the agent’s goal is unifiable with π_h and the agent believes ϕ , then π_h is replaced by π_b . The practical reasoning module of a 3APL program is a set of practical reasoning rules.

A practical reasoning rule can be applied to a goal by unifying the head of the rule with the initial part of the goal. Since goal variables may occur in the head and the body of practical reasoning rules, the unification results in a substitution for goal variables. The resulting substitution will be applied to the body of the practical reasoning rule and the resulting goal will replace the initial unified part of the goal to which the rule was applied. For example, consider the practical reasoning rule $X; B() \leftarrow \top \mid X; X$ and the goal $\pi = A(); B(); C()$. The application of the rule to π results in the substitution $[X/A()]$ which, when applied to the body of the rule, gives the goal $A(); A(); C()$. Of course, there are many different choices for how to unify the head of a rule with a goal. A discussion of this issue is out of the scope of this paper.

Practical reasoning rules are used for various reasons. They can be applied to revise agents' goals that are not achievable, to actions that are blocked, to optimize agents' goals, to generate goals, or to plan a goal. In the context of this paper we are only interested in the rules for goal generation and planning.

- **Goal generation rule:** A goal generation rule is a practical reasoning rule that has the form $\top \leftarrow \phi \mid g$, where $\phi \in \mathcal{L}_B$ and g is a predicate goal. A goal generation rule indicates that the goal g can be adopted if the agent believes ϕ . A goal generation rule will be noted as $\phi \Rightarrow g$. The set of goal generation rules is denoted by S_g .
- **Planning rule:** A planning rule is a practical reasoning rule of the form $g \leftarrow \phi \mid \pi$, where g is a predicate goal, $\phi \in \mathcal{L}_B$, and $\pi \in \mathcal{L}_G$. The goal g can be achieved by plan π if ϕ is believed. The set of planning rules is denoted by S_p .

Finally, two strict orderings are associated with each agent. The first ordering indicates the preference of the agent with respect to its goals and the second ordering indicates the preference of the agent with respect to its planning rules. In a 3APL program, these ordering are implemented by the order of occurrences of goals and practical reasoning rules respectively. In [8] an operational semantics for the 3APL language is proposed which is defined by means of a transition system. Since the semantics of the 3APL language is not the focus of this paper, we do not present it here.

3.2 Agent Definition

For the objectives of this paper, an agent is defined as a set of predicate goals, a belief base, and a set of planning rules. Furthermore, we use two orderings: one on the goals of the agent and another on its reasoning rules.

DEFINITION 5. *A 3APL agent is a tuple $\langle \Pi, \sigma, \Gamma, \succ^g, \succ^r \rangle$, where $\Pi \subseteq \mathcal{L}_G$ is a set of predicate goals, $\sigma \subseteq \mathcal{L}_B$, $\Gamma \subseteq S_p$ is a set of planning rules, \succ^g is a complete ordering on Π , and \succ^r is a partial ordering on Γ such that \succ^r is complete for each subset of Γ that consists of rules with unifiable heads.*

The ordering on the goals is needed to prioritize which goal the agent tries to achieve next. This ordering has to be complete because the agent must always be able to make a choice between two goals. The chosen goal determines which rules are applicable. Ordering between rules is therefore only necessary for rules related to the same goal.

3.3 Role Definition

Similarly to the agent definition, we define a role through its set of predicate goals, and its interaction rules which must also be ordered. For the moment we assume the interaction rules to have the same format as the planning rules of agents. Furthermore, the definition of role includes the norms associated with the role, which are split into obligations and prohibitions.

DEFINITION 6. *A role ρ is a tuple $\langle \Omega, \Lambda, \Upsilon, \Theta, \succ^g, \succ^r \rangle$, where Ω is a set of predicate goals that should be achieved when enacting the role, $\Lambda \subseteq S_p$ is a set of planning rules that specifies how goals should be achieved, $\Upsilon \subseteq S_g$ is a set of obligations that needs to be satisfied when enacting the role, $\Theta \subseteq S_g$ is a set prohibitions, \succ^g is a complete ordering on Ω and \succ^r a partial ordering on Λ such that \succ^r is complete for each subset of Λ that consists of rules with unifiable heads.*

In the context of this paper, obligations are modelled as goal generation rules and prohibitions specify goals that the role enacting agent must avoid. We are aware that there are more accurate theoretical definitions of obligations and prohibitions but the above definition suffices for the purpose of the paper. Although the formal definition of a role is very similar to that of an agent there is a fundamental ontological difference between the two concepts. Whereas an agent eventually has an operational semantics, a role consists solely of a set of patterns to be taken up by the agents fulfilling the role. So, the goals and rules that constitute a role only get an operational semantics indirectly through the agents that take up the role.

3.4 Conflicts between Goals

In the previous sections we have explained and defined agents and roles. The first step in the process of enabling agents to enact roles is to verify whether the agent is "consistent" with the role. In this section we introduce the notion of conflict between goals and formally define it. Some of the intuitions of this section are derived from work on conflicting goals in [9]. The basic notion of conflict is defined between goal-rule units that consist of one predicate goal and a set of rules. The rules in the units can be seen as planning rules for the goal.

DEFINITION 7. *Let $\Gamma \subseteq S_g \cup S_p$ and $g \in \mathcal{L}_G$ be a predicate goal. The pair $\langle g, \Gamma \rangle$ is a goal-rule unit.*

Two goal-rule units can conflict if they have conflicting goals. For example, the goal-rule units $\langle g, \emptyset \rangle$ and $\langle \neg g, \emptyset \rangle$ conflict since g conflicts with $\neg g$. Obviously, two goal-rule units may be conflicting even when their goals are not conflicting, because conflicting goals can be generated by applying the rules. For example, $\langle g, \emptyset \rangle$ can conflict with $\langle \top, \{\top \Rightarrow \neg g\} \rangle$ since applying the rule of the second goal-rule unit generates the goal $\neg g$ which conflicts with g . Two goal-rule units can also conflict when they do not contain any goals at all. For example, $\langle \top, \{b \Rightarrow g\} \rangle$ and $\langle \top, \{b \Rightarrow \neg g\} \rangle$ conflict because the involved rules generate conflicting goals g and $\neg g$. In order to define conflicts between arbitrary pairs of goal-rule units, we introduce the notion of necessary and possible subgoals of a goal relative to a set of planning rules [9].

DEFINITION 8. *Let $\alpha \in \mathcal{L}_A$ be a basic action, $\phi \in \mathcal{L}_B$ be a belief formula, $g \in \mathcal{L}_G$ be a predicate goal, $R \subseteq S_p$, and $\pi, \pi_1, \dots, \pi_n \in \mathcal{L}_G$. Let also $\text{plan}_R^i(g)$ be the body of a*

planning rule i from R that has g as the head, $\text{plan}_R^i(g) - g$ be the same as $\text{plan}_R^i(g)$ except that all occurrences of g are removed, and m be the number of rules in R that have a unifiable head with g . The necessary subgoals of goal π relative to the set of planning rules R , denoted by $\text{sg}_R^{\text{necc}}(\pi)$, can be inductively defined as follows:

- $\text{sg}_R^{\text{necc}}(\alpha) = \emptyset$
- $\text{sg}_R^{\text{necc}}(\phi?) = \emptyset$
- $\text{sg}_R^{\text{necc}}(g) = \{g\} \cup (\bigcap_{i=1, \dots, m} \text{sg}_R^{\text{necc}}(\text{plan}_R^i(g) - g))$
- $\text{sg}_R^{\text{necc}}(\pi_1; \dots; \pi_n) = \bigcup_{i=1, \dots, n} \text{sg}_R^{\text{necc}}(\pi_i)$
- $\text{sg}_R^{\text{necc}}(\text{IF } \phi \text{ THEN } \pi_1 \text{ ELSE } \pi_2) = \text{sg}_R^{\text{necc}}(\pi_1) \cap \text{sg}_R^{\text{necc}}(\pi_2)$
- $\text{sg}_R^{\text{necc}}(\text{WHILE } \phi \text{ DO } \pi) = \text{sg}_R^{\text{necc}}(\pi)$

For possible subgoals of π relative to R , written as $\text{sg}_R^{\text{pos}}(\pi)$, we need to replace \cap for the IFgoal with \cup .

A possible way to define a conflict between a pair of goal-rule units is when their goals together with their possible subgoals are conflicting. This is a very strong notion of conflict as it takes all possible subgoals into account. Alternatively, a pair of goal-rule units may conflict when their goals together with their necessary subgoals are conflicting. This is a much weaker notion of conflict as it takes only the necessary subgoals into account.

DEFINITION 9. Let $u_1 = \langle g, \Gamma \rangle$ and $u_2 = \langle g', \Gamma' \rangle$ be two goal-rule units. The strong conflict between goal-rule units u_1 and u_2 , written as $u_1 \# u_2$, is defined as follows:

$$u_1 \# u_2 \Leftrightarrow \exists h \in \text{sg}_\Gamma^{\text{pos}}(g) \exists h' \in \text{sg}_{\Gamma'}^{\text{pos}}(g') (h = \neg h')$$

The weak conflict between units u_1 and u_2 , written as $u_1 \# u_2$, is defined as follows:

$$u_1 \# u_2 \Leftrightarrow \exists h \in \text{sg}_\Gamma^{\text{necc}}(g) \exists h' \in \text{sg}_{\Gamma'}^{\text{necc}}(g') (h = \neg h')$$

The above definitions can be extended to conflicts between sets of goals. For this purpose, we introduce the notion of goalset-rule unit. A goalset-rule unit is a pair $\langle \Pi, \Gamma \rangle$ for $\Pi \subseteq \mathcal{L}_G$ and $\Gamma \subseteq S_g \cup S_p$. In the following, we consider only the strong notion of conflicts between goals. To look at other types of conflicts between goals would mean an explosion on the number of possible combinations, which would take too much space to describe and explain. The stronger notion we look at, is the most restrictive case, that is, indicates a 'real' conflict between goals that cannot be ignored.

DEFINITION 10. Let $u_1 = \langle \Pi, \Gamma \rangle$ and $u_2 = \langle \Pi', \Gamma' \rangle$ be goalset-rule units. The strong conflict between units u_1 and u_2 , written as $u_1 \# u_2$, is defined as follows:

$$u_1 \# u_2 \Leftrightarrow \forall g \in \Pi \forall g' \in \Pi' (\langle g, \Gamma \rangle \# \langle g', \Gamma' \rangle)$$

The weak conflict between units u_1 and u_2 , written as $u_1 \# u_2$, is defined as follows:

$$u_1 \# u_2 \Leftrightarrow \exists g \in \Pi \exists g' \in \Pi' (\langle g, \Gamma \rangle \# \langle g', \Gamma' \rangle)$$

Note that we could have defined two more notions of conflicting goalset-rule units by changing the quantifiers in the definition. We could also have varied by changing between strong and weak conflicting goals. Together, this would have given us 8 possible gradations of conflicts between goalset-rule units. Although interesting in themselves we have limited ourselves to the extremes defined above which seem to be the more useful (necessary) ones for the moment.

4. AGENTS ENACTING ROLES

In order to define when an agent can play a role without being in conflict with or violating a norm, we first need to consider if an agent or a role is internally coherent, i.e. if the goals and plans of an agent are conflicting or if the goals, plans, and norms of a role can conflict. Moreover, we assume that the internal coherence of an agent or a role also depends on whether goals and subgoals, associated with an agent and a role, are achievable, i.e. whether they can be planned by the planning rules associated to the agent and the role, respectively.

DEFINITION 11. An agent $A = \langle \Pi, \sigma, \Gamma, \succ^g, \succ^r \rangle$ is internally coherent if and only if:

- The goals of an agent are not conflicting, i.e.

$$\neg(\langle \Pi, \Gamma \rangle \# \langle \Pi, \Gamma \rangle)$$

- The goals of an agent can be planned and reached, i.e.

$$\forall g \in \bigcup_{g' \in \Pi} \text{sg}_\Gamma^{\text{pos}}(g') \exists b \in \mathcal{L}_B \exists \pi \in \mathcal{L}_G (g \leftarrow b \mid \pi \in \Gamma)$$

Moreover, an internally coherent agent is called effective if and only if:

- All planning rules are applicable, i.e.

$$\forall \gamma \in \Gamma (\gamma = g \leftarrow b \mid \pi \wedge g \in \bigcup_{g' \in \Pi} \text{sg}_\Gamma^{\text{pos}}(g'))$$

For a role the internal coherence means that goals, plans, obligations and prohibitions do not conflict with each other and the goals are achievable.

DEFINITION 12. A role $R = \langle \Omega, \Lambda, \Upsilon, \Theta, \succ^g, \succ^r \rangle$ is internally coherent if and only if:

- Goals of a role are not conflicting, i.e.

$$\neg(\langle \Omega, \Lambda \rangle \# \langle \Omega, \Lambda \rangle)$$

- Goals of a role can be planned and reached, i.e.

$$\forall g \in \bigcup_{g' \in \Omega} \text{sg}_\Lambda^{\text{pos}}(g') \exists b \in \mathcal{L}_B \exists \pi \in \mathcal{L}_G (g \leftarrow b \mid \pi \in \Lambda)$$

- Goals and obligations of a role do not conflict, i.e.

$$\neg(\langle \Omega, \Lambda \rangle \# \langle \Upsilon, \Upsilon \rangle)$$

- Goals and prohibitions do not conflict, i.e.

$$\forall g \in \bigcup_{g' \in \Omega} \text{sg}_\Lambda^{\text{pos}}(g') \forall b \in \mathcal{L}_B (b \Rightarrow g \notin \Theta)$$

- Obligations and prohibitions of roles do not conflict, i.e.

$$\forall g \in \bigcup_{b \Rightarrow g' \in \Upsilon} \text{sg}_\Lambda^{\text{pos}}(g') \forall b' \in \mathcal{L}_B (b' \Rightarrow g \notin \Theta)$$

Moreover, an internally coherent role is effective if and only if:

- All planning rules are applicable, i.e.

$$\forall \lambda \in \Lambda (\lambda = g \leftarrow b \mid \pi \wedge g \in \bigcup_{g' \in \Omega} \text{sg}_\Lambda^{\text{pos}}(g'))$$

Before we can discuss how an agent takes up a role we must discuss the relationship between roles and agents.

4.1 Matching agents and roles

In the following we consider two different relations between an agent and a role. The first relation is called compatibility, which is based on the subset relation between components of the agent and the role. Informally, an agent is compatible with a role if and only if the agent (sub)goals are a subset of (sub)goals of the role. That is, the agent (naturally) fulfills the goals of the role. Conversely, a role is compatible with an agent if and only if the (sub)goals of the role are a subset of (sub)goals of the agent.

DEFINITION 13. Let $A = \langle \Pi, \sigma, \Gamma, \succ_A^g, \succ_A^r \rangle$ be a 3APL agent and $R = \langle \Omega, \Lambda, \Upsilon, \Theta, \succ_R^g, \succ_R^r \rangle$ be a role. Agent A is compatible with role R iff $\Pi \subseteq \Omega$ and $\Gamma \subseteq \Lambda$. Conversely, role R is compatible with agent A iff $\Omega \subseteq \Pi$ and $\Lambda \subseteq \Gamma$.

For example, if an agent A has the goal of reading a certain paper and there is a role of *PC-member* according to which one must review that same paper, agent A is compatible with the role of *PC-member* since the subgoals of *review-paper* includes goal *read-paper*. Compatibility indicates that the agent is highly suitable to fulfill a role. However, such a match between agents and roles is not always possible. We therefore introduce a weaker relation between agents and roles. This relation, consistency, indicates that the goals of the agent and the goals of the role do not conflict.

DEFINITION 14. Let $A = \langle \Pi, \sigma, \Gamma, \succ_A^g, \succ_A^r \rangle$ be an internally coherent agent and $R = \langle \Omega, \Lambda, \Upsilon, \Theta, \succ_R^g, \succ_R^r \rangle$ be an internally coherent role. Agent A is consistent with role R if and only if:

- The goals and rules of the agent and the role do not conflict:

$$\neg(\langle \Pi, \Gamma \rangle \# \langle \Omega, \Lambda \rangle)$$

- The goals and rules of the agent do not conflict with the obligations of the role:

$$\neg(\langle \Pi, \Gamma \rangle \# \langle \Upsilon, \Upsilon \rangle)$$

- The goals and subgoals of the agent do not conflict with the prohibitions of the role:

$$\forall g \in \bigcup_{g' \in \Pi} sg_{\Gamma}^{pos}(g') \quad \forall b \in \mathcal{L}_B \quad (b \Rightarrow g \notin \Theta)$$

For example, agent A with the goal of submitting a paper is consistent with the role of *PC-member* since both goals do not 'interfere' with each other. We can now define the concept of role enabled agents.

DEFINITION 15. Given an internally coherent agent A and an internally coherent role R :

- A is strongly enabled to enact R iff A is compatible with R
- A is weakly enabled to enact R iff A is consistent with R

4.2 Taking up the role

Given an internally coherent agent A and an internally coherent role R , we can now consider what it means for agent A to enact the role R appropriately. In this paper, we consider that when an agent A enacts role R , A will adopt all

the goals and the rules associated with R . A will also include the obligations of the role as rules in its rule base. These will trigger the agent to fulfill the obligations of the role. In addition, the agent can keep some or all of its own goals and rules, as long as it keeps its internal coherence. Furthermore, in order to achieve its goals, an agent must select and plan them. This selection mechanism is based on the ordering on goals and planning rules. When A adopts the goals and rules of a role R , A must also extend its orderings to include the goals and rules of the role. Of course, this can be done in many different ways which indicate how the agent enacts the role. For instance, this ordering can give preference to the agent's own goals (a selfish agent), or to the goals of the role (a social agent). In the most simple case, such ordering will be imposed on the agent by the society. In the future, agents should be able to reason and negotiate about combining these orderings. See also [11] for some possible mental consequences for agents when taking up a role.

DEFINITION 16. Let $A = \langle \Pi, \sigma, \Gamma, \succ_A^g, \succ_A^r \rangle$ be a 3APL agent and $R = \langle \Omega, \Lambda, \Upsilon, \Theta, \succ_R^g, \succ_R^r \rangle$ be a role. The role enacting agent $R(A)$ is an agent, enabled to enact role R , such that $R(A) = \langle \Omega \cup sub(\Pi), \sigma, \Lambda \cup \Upsilon \cup sub(\Gamma), \succ_{R(A)}^g, \succ_{R(A)}^r \rangle$ where $sub(\Pi)$ and $sub(\Gamma)$ respectively represent subsets of Π and Γ , and $\succ_{R(A)}^g, \succ_{R(A)}^r$ are ordering relations on the goals and reasoning rules for $R(A)$. Besides the usual structural constraints on these orderings, they should preserve the goal and rule orderings of role R .

For example, consider agent¹ $A = \langle \{g^A\}, \sigma, \Gamma, \succ_A^g, \succ_A^r \rangle$ where $g^A = read-paper$. A is enabled to enact the *PC-member* role specified as $PC-member = \langle \{g_1^{pc}, g_2^{pc}\}, \Lambda, \emptyset, \succ_{pc}^g, \succ_{pc}^r \rangle$, where $g_1^{pc} = review-paper$ and $g_2^{pc} = announce-conference$ such that $g_1^{pc} \succ_{pc}^g g_2^{pc}$. The role-enacting agent $pc(A)$ is specified as $\langle \{g_1^{pc}, g_2^{pc}, g^A\}, \sigma, \Lambda \cup sub(\Gamma), \succ_{pc(A)}^g, \succ_{pc(A)}^r \rangle$.

According to definition 16, the ordering relation $\succ_{R(A)}^g$ must preserve the ordering of goals in the role, i.e. $g_1^{pc} \succ_{pc}^g g_2^{pc}$. How the order between goals of the role and goals of the agent is to be specified is up to $pc(A)$. For this example the following orders are possible:

$read-paper \succ_{pc(A)}^g review-paper \succ_{pc(A)}^g announce-conference$
 $review-paper \succ_{pc(A)}^g read-paper \succ_{pc(A)}^g announce-conference$
 $review-paper \succ_{pc(A)}^g announce-conference \succ_{pc(A)}^g read-paper$

Note that the ordering taken by the role enacting agent indicates the attitude of the agent to the role it is taking up.

It is clear that an appropriate enacting relation presupposes both compatibility and consistency relations. An agent that is neither compatible nor consistent with a role has apparently goals and planning rules that, when enacting the role, may violate some norms that are associated with the role. A critical case is when an agent is consistent with a role but not compatible. In such a case, the agent has apparently some additional goals or planning rules that are not associated with the role. Although this agent will not violate any norms associated with the role, the agent may use its own preference ordering to achieve its own goal preferably and thus ignore the goals that are prescribed by the role. It is clear that in such a case, the agent enacts the role inappropriately. The same situation can also occur even when

¹To save space, we only mention the elements of an agent definition that are relevant for the example.

the agent and the role are compatible. In such situations, the appropriate enacting relation is not guaranteed since the enacting relation depends on the ordering on goals and roles as well. An agent can in fact use its own preference relation and thereby never achieve the goals associated with a role.

DEFINITION 17. *Given the above discussion we can distinguish at least the following types of role enactment by the agents:*

- **selfish enactment:** *The enacting agent includes a subset of its own goals and rules, and gives priority to its own goals and rules.*
- **social enactment:** *The enacting agent includes a subset of its own goals and rules, but gives priority to the goals and rules of the role.*
- **maximally social enactment:** *The enacting agent only uses the goals and rules from the role, and adopts the orderings of the role.*

Of course it is easy to see that many more combinations could be made. The above classification is based on the assumption that the ordering on goal and rules of the enacting agent is in accordance with the ordering that is prescribed by the role. This assumption can be relaxed in which case the agent can either use its own ordering, the ordering of the role, or a combined ordering. Moreover, we may assume an ordering on the obligations and prohibitions associated to the role. In general, the possible choices to use these ordering result in a variety of agent types. We leave it for further research, which other types of enactment might be useful in practice. However, we would at least like to mention two prototypical examples of agents:

DEFINITION 18. *An agent that follows its own preference orderings rather than the orderings associated with the role is called a selfish agent. An agent that follows the orderings associated with the role rather than its own orderings is called social agent.*

Thus, in the case of conflicting orders, a social agent will adopt the order associated to the role and not its own. In contrast, a selfish agent will use its own order rather than the order associated to the role. We finish this section with the following proposition that indicates some relations between the different concepts that we have defined in the previous sections.

PROPOSITION 1. *Let $A = \langle \Pi, \sigma, \Gamma, \succ_A^g, \succ_A^r \rangle$ be an internally coherent agent and $R = \langle \Omega, \Lambda, \Upsilon, \Theta, \succ_R^g, \succ_R^r \rangle$ be an internally coherent role such that $\succ_A^g \subseteq \succ_R^g$ and $\succ_A^r \subseteq \succ_R^r$. Then,*

- *A is compatible with R implies A is consistent with R.*
- *If A and R are effective and A is consistent with R, then the enacting agent $R(A)$ is internally coherent and effective.*
- *If A is compatible with R and A is a social agent, then agent $R(A)$ enacts role R maximally social.*
- *If A is consistent with R and A is a social agent, then the enacting agent $R(A)$ enacts R socially.*

We leave the proofs out due to space limitations. Most of them follow directly from the definitions given before.

5. FURTHER RESEARCH

Ideally the paper would continue with a section on how the agent exits from a role. It is obvious that an agent should not be allowed to exit a role at any time the agent wishes to do so. Enacting a role creates the expectation that the agent will fulfil the goals and norms associated with the role, and an agent should not exit the role before those goals are achieved, or otherwise it should pay some penalty. Some of this notions could be modelled by specifying so-called exit states/goals of the role. That is, an agent can exit the role whenever it fulfills one of these pre-defined goals. Of course, this raises the question whether an agent cannot exit a role anymore when it genuinely tried to fulfill the goals, but is not capable of doing so (due to special circumstances).

Another question that has to be answered is the state of the agent after it exits the role. If the agent had given up its own goals and rules for those of the role it enacted, then it has to recover them somehow. Also it might have to return to a particular goal it tried to fulfill before taking up the role. However, it should then determine how the role enactment contributed to the fulfilment of that goal in order to check which subgoal to pursue next. In an extreme case the role enactment was forced upon the agent and can be considered as a kind of interrupt after which the agent can continue where it left of. Normally, however, enacting the role served some purpose for the agent and it should check whether the purpose was (partially) reached. It might be clear that these issues still need thorough investigation and cannot be dealt with in the little space available at this point.

Another point for further research concerns the types of relations between agents and roles. In this paper the notion of role enacting agents is based on the assumption that the agents and the roles that they enact are not conflicting. Clearly, this is a strong limitation for agents to enact roles. In general we believe that agents should be able to enact a role even when they are conflicting. However, agents will probably not fulfill the role optimally or efficiently when they conflict with the role. An interesting issue is in this case the level of role fulfillment when certain conflict exists. Therefore, possible conflicts and their properties should be identified and classified. For each class of conflicts the level of role fulfillment should be determined. Another related issue is the partial fulfillment of a role by an agent. The question is which parts of a role can be fulfilled by an agent that conflict with the role.

Concerning the compatibility relation itself there are also some areas for further exploration. Up to now the compatibility relation was defined based on set inclusion and identity. However, we may extend the notion of compatibility by considering goals and plans that are not based on identity and inclusion. We may also consider the case where the agent not only adds goals and rules to its components, but it also adds basic actions in order to be enact a relation.

Finally, there are some areas for further research related to the specification of the agents and roles themselves. The notion of goal as discussed in this paper is a kind of to-do-goal, which is related to the notion of task or plan. These goals do not denote states and cannot be composed by means of logical connectives. As a consequence, the agent can reason about goals only by means of practical reasoning rules and not by means of logical relations between goals. We are developing an extension of 3APL in which the concept of to-be-goal is available. In order to realize this extension we

consider two sets of state formulae. The first set contains the belief formulae, which denote the current state of the agent. The second set of state formulae contains the goal formulae, which denote the states to be reached. Of course, we demand that a goal formula should not be a belief formula since such a case indicates that a goal (a to-be-reached state) is already reached. Using this notion of goal, most of the proposed definitions in this paper should be reformulated and explained in terms of goal formulae. The preference of agents can then be defined in the standard way in terms of the orderings on states.

Another logical extension of the agent specification is of course adding norms to the agents themselves. Normative agents will have their own set of norms that should be compared to the goals and norms of the role in order to check whether the agent is consistent with the role and is likely to fulfill it in a satisfactory way.

6. CONCLUSIONS

Given the importance of open agent systems and the fact that in these systems agents will no longer be designed for one particular role, but are rather developed independently and have their own goals and rules, it becomes imperative to design mechanisms to check the compatibility of certain agents with roles in the agent system. In this paper we have presented on going research on the determination of some necessary conditions under which an agent can enact a role and what it means for the agent to enact a role. We have shown that agents can be more or less 'suitable' to fulfil a role depending on the relation between an agent's own goals and the requirements of the role. Besides the suitability of an agent, there are also a number of ways an agent can enact the role. For example, whether the role gets priority or not. Each of those choices can be represented as a specific architectural adaptation of the agent. An important point for further research is to formally describe how an agent can exit the role.

7. REFERENCES

- [1] A. Artikis and J. Pitt. A formal model of open agent societies. In *Proc. Autonomous Agents 2001*, pages 192–193, 2001.
- [2] L. Cavedon and L. Sonenberg. On social commitments, roles and preferred goals. In *Proc. of the Third Int. Conference on Multi-Agent Systems (ICMAS-98)*, pages 80–87. IEEE Computer Society, 1998.
- [3] M. Dastani and L. van der Torre. What is a normative goal? In *Proceedings of International Workshop on Regulated Agent-Based Social Systems: Theories and Applications (RASTA'02)*, AAMAS'02 Workshop, Bologna Italy, 2002.
- [4] C. Dellarocas and M. Klein. Civil agent societies: Tools for inventing open agent-mediated electronic marketplaces. In *Proc. AMEC (at IJCAI'99)*, Stockholm, Sweden, 1999.
- [5] V. Dignum, J.J. Meyer, H. Weigand, and F. Dignum. An organizational-oriented model for agent societies. In *Proc. RASTA (at AAMAS'02)*, Bologna, Italy, 2002.
- [6] M. Esteva, J. Padget, and C. Sierra. Formalizing a language for institutions and norms. In *Proceedings of the 8th International Workshop on Agent Theories, Architectures and Languages, ATAL-2001*, Seattle, 2001.
- [7] M. Fasli. On commitments, roles and obligations. In B. Dunin-Keplicz and E. Nawarecki, editors, *From Theory to Practice in Multi-Agent Systems*, volume 2296 of *LNAI*, pages 93–102. Springer-Verlag, 2002.
- [8] K. Hindriks, F. de Boer, W. van der Hoek, and J.J. Meyer. Agent programming in 3APL. *Autonomous Agents and Multi-Agent Systems*, 2(4):357–401, 1999.
- [9] L. Padgham J. Thangarajah, M. Winikoff and K. Fischer. Avoiding resource conflicts in intelligent agents. In F. van Harmelen, editor, *Proceedings of the 15th European Conference on Artificial Intelligence (ECAI-2002)*. IOS Press, 2002.
- [10] N. Jennings, K. Sycara, and M. Wooldridge. A roadmap for agent research and development. *Autonomous Agents and Multi-Agent Systems*, 1(1):7–38, 1998.
- [11] J. Sichman and R. Conte. On personal and role mental attitudes: A preliminary dependence-based analysis. In F. Oliveira, editor, *Advances in AI*, volume LNAI-1515, pages 1–10. Springer-Verlag, 1998.
- [12] W. Vasconcelos, J. Sabater, C. Sierra, and J. Querol. Skeleton-based agent development for electronic institutions. In *Proc. UKMAS'01*, 2001.
- [13] M. Wooldridge, N. Jennings, and D. Kinny. The gaia methodology for agent-oriented analysis and design. *Autonomous Agents and Multi-Agent Systems*, 3(3):285–312, 2000.