Organizations and Normative Agents

Mehdi Dastani, Virginia Dignum, and Frank Dignum

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

Abstract. The overall behavior of multiagent systems can be controlled by designing agent organizations and allowable interactions between autonomous and goal directed agents such as 3APL agents. The rules of behavior for individual agents are described using concepts from organization theories such as roles and norms. In this paper, a framework for agent organization is discussed and its related concepts such as norms and roles are formalized. It is argued that organizational roles and norms influence and determine the goals of individual agents whenever they are involved in the corresponding organization. Some formal properties of the relation between individual agents and their organizations is presented.

1 Introduction

Software agents and in specific Multi-agent systems [5, 6] are one of the most promising areas in the field of computer science. Software agents get some knowledge about the world in which they operate, such that they can solve most of the minor problems they encounter in operation by themselves, without intervention of the user. This has a large advantage over traditional systems for which the environment of the system had to be completely predictable or otherwise the system would not function correctly. This might seem trivial, but if I am a user that only know that there exists a handy program for calculating the contents of a cilinder, but do not know whether I should give it the radius or diameter of the cilinder as argument I cannot use the program. Especially in open environments such as the Internet or Intranets agents will be able to react more flexible and cooperatively with other systems than traditional software.

However, the autonomy of the agents also has a downside. If one creates a system with a number of autonomous agents it becomes unpredictable what the outcome of their interactions will be. This so-called emerging behavior can be interesting in settings where the multi-agent system is used to simulate a group of people and one tries to find out which factors influence the overall behavior of the system. E.g. some studies have been done in which groups of selfish agents are compared with cooperative agents. (In general the system with cooperative agents produces better results for the individual agents).

However, in settings where the multi-agent system is used to implement a system with a specific goal one does not want this emergent behavior to diverge from the overall goal of the system. E.g. if the system is designed to get up-to-date information about the indexes of the stock exchanges in London, Frankfurt and new York, one does not want the system to evolve into a system that gives up-to-date information about bonds in Tokyo and Sydney (even though it still gives financial information). In order to limit the autonomy of the agents in these situations and ensure a certain behavior of the overall system we have to design agent organizations and allowable interactions between the agents in these organizations. The rules of behavior for the agents within the organization are described using "norms". Of course it is important that we also define how the agents use these norms to govern their behavior as this determines the interaction between the individual agents and the multi agent system. In this paper we will explore this interface between the individual agent with its autonomy and the society with its norms.

In the next section we will describe multi-agent systems from the organizational point of view and indicate which choices should be made and what concepts are used to describe this view. In section 3 a model to design organizational multi-agents systems is described. In section 4 we will introduce 3APL, an agent programming language that is used to specify the behavior of certain types of individual agents. In section 5 we will describe the mapping of the social aspects of the agent organization into the description of the behavior of the individual 3APL agents. An example of such a mapping is given in section 6.

2 Agents Societies

The design of multi-agent systems must consider organizational characteristics such as stability over time, some level of predictability, and commitment to aims and strategies. The development of multi-agent systems calls for models, languages and methodologies to represent communication, interaction, roles and other concepts that characterize multi-agent systems. Such modelling primitives are usually not provided by (single) agent languages. Furthermore, traditional multi-agent models and architectures often assume an individualistic perspective in which agents are taken as autonomous entities pursuing their own individual goals based on their own beliefs and capabilities. In this perspective global behavior emerges from individual interactions and cannot easily be managed or specified externally.

Agent societies are an effective platform to model organizations because they provide mechanisms to allow organizations to advertise their capabilities, negotiate their terms, exchange rich information, and synchronize processes and workflow at a high-level of abstraction. From an organizational perspective, it is the society goals that determine agent roles and interaction norms. In agent societies, individual agents are therefore seen as actors that perform role(s) described by the society design; they interact with each others to accomplish their goals by playing their roles. However, the agent's own capabilities will determine the specific way an agent enacts its role(s). Therefore, frameworks for agent societies must combine models that describe the structure and characteristics of an organization with models that specify the interests and capabilities of the involved individuals.

In [2], we proposed a framework for agent societies that incorporates organizational and individual perspectives as described above. The framework consists of three interrelated models each describing different aspects of the society: organizational model, agent model and social model. We assume that individual agents are designed independently from the organization. The social model provides a dynamic link between agents and organization. Organizational model describes the desired or intended behavior and the overall structure of the society. This model does not include or refers to agents, but only to roles, which are described in terms of externally perceived actions and behavior. Agent model is used to describe the agents that will participate in the society, in terms of their capabilities, goals and interaction patterns. Social model specifies the relation between organizational roles and specific agents. In this model, the organizational model is populated by agents that fulfil the designed roles and interact according to the defined rules.

3 Organizational Model

The organizational model specifies the structure of an agent society according to the requirements of the organization itself. In its most simplified form, a society can be defined in terms of its objectives (goals) and the norms that regulate interaction in the society. The goals of a society are specified in terms of roles that correspond to the different stakeholders in the domain. That is, an overall goal of a society is represented as an hierarchy of subgoals that correspond to the goals of the different roles. A role is defined as a triple $< G, N, R >$, where G, N, and R stand respectively for the goals, the norms, and the interaction rules that are associated with the role. Furthermore, the organizational model is split into two parts: facilitation and operation. The facilitation layer provides the backbone of the society and consists of institutional agent roles, which are designed to enforce the social behavior of agents in the society and assure the global activity of the society. The operational layer models the overall objectives and intended action of the society and consists of domain related roles.

For example, consider the case of a trading society. The overall goal of the society is to generate transactions. This goal can be split into a facilitation component that aims at the regulation of those transactions, and an operational part where transactions are generated. Typical domain related and operational stakeholders in such a society are sellers and buyers, which exhibit autonomous behavior in the society. These will be specified as roles in the operational component of the model. Finally the activity of the facilitation layer can be described in terms of a registrar role that regulates the participants and a market master role that regulates the transactions and supports the matching between sellers and buyers.

The roles at the lowest level can be described as follows. The (S)eller represents an entity that wants to exchange its goods for money; the (B)uyer rep-

Fig. 1. The role interactions in the organization.

resents an entity that wants to exchange its money for goods; the (M)arket master takes care of introducing potential sellers to potential buyers; and the (R)egistrar keeps track of who are the sellers and buyers at each moment. The interaction between these roles are depicted in figure 1. The roles can be specified by considering the norms that regulate these roles in terms of its obligations (O), permissions (P) and prohibitions (F). In this example, the role *Registrar* has the obligation to register buyers and sellers and to announce new buyers and sellers to the Market Master, the role *Market Master* has the obligation to announce goods for sale to Buyers and is forbidden to register as buyer or as seller. The role *Buyer* is obliged to register as buyer and to pay for the bought goods, and is permitted to make offers to sellers. Finally, the role *Seller* is obliged to register as seller, to register goods to sell with Market master, to supply sold goods to buyers, and to indicate the sold goods to the Marker master. The way actions are performed can be described in interaction protocols, e.g. *pay goods* can be implemented as sending a credit card number.

4 Agent Model

In this section, we consider a particular model for social agents called 3APL. These agents are cognitive agents that have beliefs and goals as mental attitudes and can revise or modify their goals by means of the so-called practical reasoning rules. Moreover, they are assumed to be capable of performing certain basic actions such as sensing and communication actions as well as mental updates. The specification of 3APL is introduced in [4]. A 3APL agent can be specified in terms of its beliefs, goals, and goal revision rules. These rules are called practical reasoning rules.

Definition 1 (Beliefs). *Given a set of domain variables VAR and functions (constants are zero-place functions), the set of domain terms is defined as usual. Let* $t_1, \ldots, t_n$ *be terms referring to domain elements and* $Pred$ *be a set of domain predicates, then the set of programming constructs for belief formula,* BF*, is defined as follows: If* $p \in Pred$*, then* $p(t_1, \ldots, t_n) \in BF$*, and If* $\varphi, \psi \in BF$*, then* $\neg\varphi, \varphi \wedge \psi \in BF$*. All variables in the* BF *formula are universally quantified with maximum scope.*

Agents are assumed to be able to perform a set of basic actions. These actions, which are also called basic capabilities, are parameterized actions and are defined in terms of pre- and post-conditions.

Definition 2. *Let t_i be a term denoting a domain element, A_i be an action symbol and $\phi, \psi \in BF$. Then, $\{\phi\}A_i(t_1, \ldots, t_n)\{\psi\}$ is a parameterized basic action with ϕ and ψ as pre- and post-conditions, respectively. The set of basic actions is denoted by $Bact$.*

Goals in 3APL can be best understood as partial plans which should be performed by the agents. The goals are based on action operators, primitive actions, or other plans. We also assume a set of goal variables, $GVAR$ ($GVAR \cap VAR = \emptyset$) that can be used to refer to unidentified goals.

Definition 3 (Goals). *Let $Bact$ be a set of basic actions, BF be a set of belief sentences, $\varphi \in BF$, and $\pi_1, \pi_2 \in G$. Then, the set of 3APL goals (G) are defined as follows: $GVAR, Bact, BF \subseteq G$, and $(\varphi?)$, $(\pi_1; \pi_2)$, $(\pi_1 + \pi_2)$, $(\pi_1 || \pi_2) \in G$.*

Practical reasoning rules are at the heart of the functioning of 3APL agents. For each goal of the agent, that is not a basic action, there should be a practical reasoning rule indicating how the goal can be achieved. However, the practical reasoning rules can also be applied to revise agent's goals that are not achievable, basic actions that are blocked, to optimize agent's goals, to generate some sort of reactive behavior, or to define achievement goals (i.e. procedures).

Definition 4 (Practical Reasoning Rules). *Let $\pi_h, \pi_b \in G$ and $\varphi \in BF$, then a practical reasoning rule is defined as follows: $\pi_h \leftarrow \varphi \mid \pi_b$. This rule can be read as follows: if the agent's goal is π_h and the agent believes φ, then π_h can be replaced by π_b.*

The 3APL agent functions as follows: at each cycle the agent checks what its current goal is and tries to execute the goal if it is a basic action or find a practical reasoning rule whose head can be unified with the current goal and whose guard is true. If the agent succeeds in finding such a rule, it rewrites the current goal to the body of the rule (given the unification of the variables in the goal and the rule). This process continues until the agent has no more goals left or no rule is applicable. Given the definition of beliefs, goals and practical reasoning rules, a 3APL agent can be specified as follows:

Definition 5. *A 3APL agent is a triple $< \Pi, \sigma, \Gamma >$, where Π is a set of goals, σ is a set of belief formula, and Γ is a set of practical reasoning rules.*

For example, let $\{\neg p(a)\}$ $A()$ $\{p(a)\}$ and $\{p(a)\}$ $B()$ $\{\neg p(a)\}$ be two basic actions. Then, $< \ \Pi = \{A(); B()\}$, $\sigma = \{q(b)\}$, $\Gamma = \{A(); X \leftarrow p(a) \mid X; A()\} \ >$ is an agent which has a goal to do first $A()$ and then $B()$, believes $p(a)$, and has a goal revision rule which states that whenever it has to do $A()$ and after that something else, but believes $p(a)$ (i.e. the precondition of $A()$ is not satisfied), then it delays the execution of $A()$ and does X first.

5 Social Model

A trivial characterization of the social model for goal directed agents such as 3APL is to relate the goals and interaction rules that are associated with the organizational role to the goals and practical reasoning rules of role playing 3APL agents. Although this characterization seems to be intuitive, it does not characterize the interaction between role-related goals and practical reasoning rules with the individual goals and practical reasoning rules of the agent itself. Moreover, this characterization does not explain the relation between norms that are associated with the organizational roles and the model of 3APL agents.

In order to characterize the relation between role-related and individual goals and practical reasoning rules, the specification of agents can be extended to include, besides goalbase Π and practical reasoning rules Γ, two new components called normative goalbase and social interaction rules, respectively. The first component contains role-related goals and the second component contains role-related interaction rules.

Definition 6. *A social 3APL agent is a 5-tuple* $< \Pi, \sigma, \Gamma, \Delta, \Upsilon >$ *where* Π *is the set of agent's goals,* σ *its beliefs,* Γ *its practical reasoning rules,* Δ *its normative goals, and* Υ *its social interaction rules.*

The reason to distinguish between individual and role-related goals and rules is to indicate goals and rules that the agent should pursue when the agent plays the role. The social model, i.e. the relation between individual and role-related goals and practical reasoning rules, can be characterized by the following definition.

Definition 7. *Let* G_ρ *and* R_ρ *be goals and interaction rules associated with the organizational role* ρ*. A 3APL agent* $\alpha =< \Pi, \sigma, \Gamma, \Delta, \Upsilon >$ *plays the organizational role* ρ *if it satisfies the following property:* $G_\rho \subseteq \Delta \ \wedge \ R_\rho \subseteq \Upsilon$*.*

Moreover, we allow agents to aim at achieving their individual goals, i.e. the goals that are not associated with its role but are compatible with it. We assume that Δ and Υ contain compatible goals and interaction rules.

Definition 8. *Let* ϕ, ψ, Φ*, and* Ψ *be respectively a goal, a rule, a set of goals and a set of rules. Let also* $compatible_g(\phi, \Phi)$ *indicate that* ϕ *is compatible with* Φ *and* $compatible_r(\psi, \Psi)$ *indicate that* ψ *is compatible with* Ψ*. A 3APL agent* $\alpha =< \Pi, \sigma, \Gamma, \Delta, \Upsilon >$*, which plays the organizational role* ρ*, satisfies the following property:* $\forall \pi \in \Pi (compatible_g(\pi, \Delta) \leftrightarrow \pi \in \Delta) \ \wedge \ \forall \gamma \in \Gamma (compatible_r(\gamma, \Upsilon) \leftrightarrow \gamma \in \Upsilon)$*.*

We can now demand that agents should aim at achieving their normative goals and apply their social interaction rules when they are playing the corresponding role. This demand does not mean that the agents cannot aim at achieving their non-normative goals since definition 8 allows agents to import some of their non-normative goals to their normative goalbase.

Proposition 1. *In an organization, agents aim at achieving their non-normative goals if the goals are compatible with their organizational roles.*

This proposition follows immediately from definitions 7 and 8.

The relations between norms and goals and interaction rules are discussed in [1]. According to this study, norms are related to goals in two ways. First, one may consider norms as a source of goals or, in other words, goals are generated by norms. Thus, when for example an agent participates in a social setting, the corresponding norms determine the goals of the agent. This type of goals are called normative goals. Second, norms may be considered as a filter or selection mechanism on agent's goals, or in other words, goals that are incompatible with norms are filtered out. Thus, when an agent plays a role, the agent should aim at achieving only those goals that are compatible with the norms associated with the role.

The goals generation view of norms can be expressed in 3APL by means of reactive practical reasoning rules. Note that the head of reactive rules are assumed to be empty. Such a rule has the form $\leftarrow \phi \mid \pi$, where the guard ϕ indicates the belief condition of the agent which needs to hold to generate the goal π; this condition should indicate the situation in which the agent can play a role. Note that reactive rules do not revise any existing goal, but rather introduce or generate new goals. The generated goals from these reactive rules will be included in the normative goalbase component of the agent.

Definition 9. *Let $\alpha =< \Pi, \sigma, \Gamma, \Delta, \Upsilon >$ be an agent that plays the role $\rho =< G, N, R >$ and I be the set of reactive rules that corresponds with norms in N. The agent α plays the role ρ if $I \subseteq \Upsilon$.*

The second view of norms, the goal selection view, can be implemented in 3APL in various ways. One way to do this is to require that agent's non-normative goals can be imported to the agent's normative goalbase only if they are compatible with the norms imposed by the social setting. This means that this aspect of norms specifies the notion of compatibility relation used in definition 8, which can be defined in various ways. For example, two goals π and π' can be defined as compatible if they are identical, if they have a subgoal relationship, i.e. $\pi = \pi_1 \cdots \pi' \cdots \pi_n$, or if one is an instantiation of the other. We will not explore this aspect of norms in more detail.

6 Example Revisited

The relevant parts of a 3APL agent that plays the registrar role, as explained in the example of section 3 can be formulated as follows:

$\Delta = \{register()\}$
$\Upsilon = \{register() \leftarrow \top \mid$
$(register\_as\_seller(Seller, Prod\_id) \; ; \; announce\_member(Seller) +$
$register\_as\_buyer(Buyer, Prod) \; ; \; announce\_member(Buyer));$
$register()\}.$

The goal *register*() is assumed to be the role-related goal of the registrar. The rule in Υ indicates that the registrar should either register a seller and announce him to the master or do it for a buyer. This activity continues by keeping the goal *register*() at the end of this rule (recursive call). In this example,

$received(\alpha, \beta, F, \phi)$ is a belief formula indicating that agent β (receiver) has received message ϕ from agent α (sender); F indicates the modality of the message (also called speech act [3]) such as $request$, $inform$, or $agree$. Finally, the goals, such as $register\_as\_seller$, are basic actions defined as follows:

1) $\{received(Seller, self, request, register\_as\_seller(Seller, Prod\_id))\}$
$register\_as\_seller(Seller, Prod\_id)$
$\{seller\_agent(Seller, Prod\_id)\}$

2) $\{received(Buyer, self, request, register\_as\_buyer(Buyer, Prod))\}$
$register\_as\_buyer(Buyer, Prod)$
$\{buyer\_agent(Buyer, Prod)\}$

3) $\{\neg announced\_to\_master(Agent), seller\_agent(Agent, P) \lor buyer\_agent(Agent, P)\}$
$announce\_member(Agent)$
$\{announced\_to\_master(Agent)\}$

7 Conclusion

In order to profit from the flexibility of autonomous agents while at the same time ensuring the robustness of the complete system, one needs to define an organizational structure for the multi-agent system. This organizational structure defines the goal of the overall system and indicates the behavioral boundaries for the individual agents within the system. Because agents come with their own goals and capabilities, we need to specify how the agent fits into a certain role it takes in the multi-agent organization. In order to implement this fitting between agents and roles in an organization we added a social component to the agent architecture. This component takes care that the goals of the agent fit within those for the role it plays and that the agent fulfills all norms belonging to the particular role. We have shown how this can be implemented in systems for which the agents are specified in 3APL which is a typical goal-directed agent language. Due to space limitations we could not explore all possible relations between the agents' goals and capabilities and those required by the role the agent plays. We hope to fully explore this part in subsequent research.

References

1. C. Castelfranchi, F. Dignum, C. Jonker, and J. Treur. Deliberate normative agents: Principles and architecture. In *Proceedings of ATAL-99*, Orlando, FL, 1999.
2. V. Dignum, J.J. Meyer, and H. Weigand. Towards an organisational model for agent societies using contracts. In *Proceedings of AAMAS'02, To appear*, Bologna, Italy, 2002.
3. FIPA. http://www.fipa.org/.
4. 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.
5. G. Weiss, editor. *Multiagent systems.* MIT, 1999.
6. M. Wooldridge and N. R. Jennings. Intelligent agents: Theory and practice. *Knowledge Engineering Review*, 10(2), 1995.