Specifying Multiagent Organizations

Leendert van der Torre[1,2]*, Joris Hulstijn[3], Mehdi Dastani[3], and Jan Broersen[3]

[1] CWI Amsterdam
[2] Delft University of Technology
[3] University of Utrecht

Abstract. In this paper we investigate the specification and verification of information systems with an organizational structure. Such systems are modelled as a normative multiagent system. To this end we use KBDIO$_{\text{CTL}}$, an extension of BDI$_{\text{CTL}}$ in which obligations and permissions are represented by directed modal operators. We illustrate how the logic can be used by introducing and discussing various properties of normative systems and individual agents which can be represented in the logic. In particular we discuss the enforcement of norms.

1 Introduction

Normative computer systems are computer systems which involve obligations, prohibitions and permissions [19]. The traditional applications can be found in computer security, for example to regulate access to file systems or libraries. Other applications have been studied in electronic commerce, in legal expert systems and in databases. See [28] for a survey on these applications. More recently, normative systems have been used to regulate virtual communities in the context of the (semantic) web. To support the development of such systems, several agent architectures have been proposed that incorporate obligations, prohibitions and permissions.

In this paper we investigate the formalization of regulations such as the widely discussed library regulations, parking regulations, copier regulations, cottage regulations, et cetera. Such examples are characterized by sometimes complicated normative systems, as well as organizational structures. Moreover, in contrast to earlier investigations, we not only consider the case in which humans interact with a normative computer system, but we also consider cases in which computers interact with other computer systems, that is, we consider multiagent systems. In particular, we consider the formalization of properties involving normative multiagent systems in an extension of Schild's BDI$_{\text{CTL}}$ [23, 11, 12], which is itself a variant of Rao and Georgeff's BDI$_{\text{CTL}}$ [21]. Such an extension consists of an extension of the logic and an extension of the properties expressed in the logic. Obligations are motivational attitudes, just like desires, but they are also related to organizational issues.

First, obligation is formalized as a directed modality [17, 13, 25, 4, 27]. Thus, whereas we may say that agent a desire to prepare a report, we say that the agent a is obliged to prepare a report *towards another agent* b. Moreover, as explained in more detail in Section 5, whereas desires and intentions remain in force as motivational attitude until the

* Supported by the ArchiMate research project.

agent *believes* they have been achieved or are no longer achievable, obligations remain in force until the agent *knows* they have been fulfilled or they are no longer achievable. We introduce an extension of $\mathrm{BDI}_{\mathrm{CTL}}$ called $\mathrm{KBDIO}_{\mathrm{CTL}}$, that makes the distinction between desires and obligations explicit, as well as the distinction between beliefs and knowledge.

Second, we provide organizational concepts such as roles, role relations, and groups in order to specify inter-agent relations that hold in organizations. The organizational concepts are interpreted as follows.

A *role* is a set of related constraints that should be satisfied when an agent enacts the role. For example, the role of project manager puts constraints on the expertise, capabilities, responsibilities, goals, obligations and permissions of the agent that enacts the role. Note that various different definitions of the concept of a role have been proposed. Our definition follows [14, 7, 29, 10]. The definition of a role is always related to some organizational activity, which determines its scope. For example, the role of chairman only makes sense during a meeting. Agents may only enact a role provided they are *qualified,* i.e., meet the basic requirements for the role.

Role relations, also known as *dependencies* or *channels*, are constraints put on a relation between roles. Examples of a role relations are *supervisor_of* and the producer-consumer relation. Role relations coordinate the behavior of different agents, similar to the way channels coordinate components in software architectures [1]. One role can be enacted by many agents. Consider for example several postmen in a district. Moreover, one agent can enact many roles. Consider for example a lecturer who is also a conference reviewer.

A *group* is a set of roles that share a group characteristic. For example, roles involved in selling goods in an organization form a group often called the selling department.

The motivation of our work is to develop a specification and verification language for normative multiagent systems with organizational structure. We therefore focus on properties of regimentation, which formalize whether norms can be violated, on deadlines, and on definitions of organizational structure. Due to the fact that we not only consider humans, but also artificial agents interacting with normative computer systems, new issues and properties arise.

For example, for agent systems it is common practice to design agents that cannot violate norms, or agents that are benevolent and will always first try to fulfill the obligations or goals of other agents, before trying to achieve their own desires. Therefore it is useful to have a specification language that can express such properties too. As these properties cannot be programmed in human agents, such properties have not made sense previously, and consequently we believe that they have not been addressed in the literature. We acknowledge the criticism on such properties, but such criticism is beyond the scope of this paper.

The layout of the paper is as follows. In section 2 we describe an example specification domain. In Section 3 we extend Schild's logic with obligations, prohibitions, permissions and organizational concepts. In the remainder, we discuss properties which can be expressed in the logic, and which can be used to specify the running example.

2 Multiagent organizations: the running example

In this section we exemplify the type of specification properties we are interested in. Shorter specification examples in subsequent sections will also apply to the domain described here. Our example domain is concerned with the different ways an organizational norm can be implemented in a multiagent system. A multiagent system developer has a choice of options to operationalize a norm. In each case, a number of assumptions about the mental attitudes and reasoning capabilities of the subjects of the norm, the individual agents, are necessary. A system developer can leave it up to the individual agents to respect the norm. In that case, he assumes that agents are benevolent or norm-abiding, and proving that the system comforms to the norm presupposes that this assumptions is formalized. By contrast, the system developer can hardwire the norm into the environment, making it physically impossible for agents to violate it. In that case, no additional assumptions on agents are needed. We believe that a rich logic like KBDIO$_{\mathrm{CTL}}$is suitable to express this kind of notions and assumptions.

The example is derived from an observation concerning different ticket policies of public transport networks [15]. Suppose ticket policies are specified as a multi-agent system. Using these specifications, one can formulate the consequences of such policies as logical properties, and verify them with respect to the system specifications.

Compare the Paris metro with a French train. On the entrance of a platform of the Paris metro, the authorities have placed a high barrier with gates that will only open when a valid ticket is inserted. Without a valid ticket, it is physically impossible to pass the barrier and use the metro. By contrast, it is possible to board a French train without a ticket. The authorities rely on personal benevolence, on social pressure, and on a sanctioning system of ticket inspection and fines, to persuade passengers to buy a valid ticket. Looking at other travel systems we find yet other solutions to the same problem: under which assumptions can we conclude that all passengers will pay for the ride? We can phrase the norm as follows:

> *When travelling by public transport, one should have paid for the trip.*

This norm is an instance of a much more general pattern occurring in situations in which humans interact with normative computer systems, and also in multiagent systems such as virtual communities or web services. For example, an agent has to access a resource offered by another agent. To regulate such access, there is an organizational structure, that may contain roles, but also more complicated normative constructs such as authorization and delegation mechanisms. In this paper we restrict ourselves to the norm above.

We consider the following ways to implement this norm in a multi-agent system. Each possibility relies on some specific assumptions about the environment, or about the agents inhabiting the system.

1. **Implementing a norm in the environment**. The norm is enforced with gates on the platform. No assumptions on the mental attitudes of agents are needed, only assumptions about their physical ability.

2. **Implementing a norm by designing benevolent or norm abiding agents**. All agents can be designed to be sincere. If they tell you they have paid, you can trust them. This removes the need for tickets as evidence of payment. Moreover, agents can be designed to be either benevolent, or norm abiding. If a benevolent agent understands why the norm is a good norm, for example to maintain a good quality of public transport, it will internalize the norm and make it a personal goal. A norm abiding agent will simply obey the norm, no matter how this relates to its own goals.
3. **Implementing a norm by relying on rationality**. Here tickets are introduced as evidence of payment, and hence as a right to travel. No sincerity assumption is needed. If an agent is caught travelling without a valid ticket, it is subject to a sanction: to pay a fine. This assumes that agents are rational decision makers, in the economic sense of maximizing expected utility. An agent will display the behavior corresponding to the norm, if a ticket is cheaper than the fine multiplied by the perceived chance of being caught. Authorities can affect this way of decision making by increasing the fine, or by making the agents believe that the chance of being caught has increased.
4. **Implementing a norm by relying on social control**. Here again tickets are used as evidence of payment. Being caught without a ticket leads to social embarrassment and a loss of reputation. Like in item 3 above, this solution assumes that agents are subject to sanctions, and moreover, that embarrassment counts as an effective sanction. Embarrassment typically only comes up if all other passengers can observe that the passenger does not pay.
5. **Implementing a norm by relying on a combination of mechanisms**. In most actual situations a mixture of these types of norm enforcement is in place. For example, a fine system is used to remind agents of the noble purpose behind the norm. Social embarrassment comes on top of the fine. That means that in practice, fines do not have to be as high as would be required for socially unaffected citizens.

Note that the above categories not only occur in human society, but also in multiagent systems. Implementing a norm in the environment is also the typical case used in web services: if an agent has not paid for the service, it simply cannot access it. Implementing a norm by norm abiding agents is not possible in human organizations, but frequently occurs in multiagent organizations. Human and multiagent systems often depend on rationality, for example in the context of electronic commerce. Finally, many human organizations rely on social control, and there are examples of multiagent systems containing social agents [8].

Obligations are motivational attitudes, just like desires, but they also have organizational aspects. First, obligations are always directed. Obligations can be directed towards abstract entities like ‘the company’ or ‘the system’, towards other agents, or towards the agents themselves. Second, the organizational structure is represented by the sets of roles, groups, and their interactions, as indicated above. Group membership and the assignment of agents to roles changes over time, as role relations are established or disconnected. The ‘social fact’ of an agent enacting a role is distinguished from the satisfaction of the requirements that go with the role. For example, although a passenger does not have a ticket while he is in the metro, even if he does not satisfy the requirements set by the role, he remains a passenger.

3 KBDIO$_{\text{CTL}}$, a logic for specifying multiagent organizations

We use a version of BDI$_{\text{CTL}}$ presented by Schild [23], which we extend with operators for knowledge and directed obligation. The syntax of KBDIO$_{\text{CTL}}$involves a modal operator K_a for knowledge of agent a, an operator B_a for belief, D_a for desire, I_a for intention, and $O_{a,b}$ for an obligation of agent a towards agent b [17, 13, 4, 27]. Knowledge, belief, desire and intention are internal to the agent and thus not directed. The temporal operators of the language are imported from CTL. To specify organizational structure, special propositions '$g(a)$', '$r(a)$', and '$a\ ch\ b$' are introduced for 'agent a is a member of group g', 'agent a enacts role r', and 'agent a and b stand in role relation ch', respectively. Higher order relations can be defined analogously. We assume that roles, groups and role relations are all primitive, though in certain systems they have been defined in terms of each other. For example, a group can be defined as the role of being a member of the group. Also, a group can be defined as a role relation between all members of the group, or between the group members and the group leader.

Definition 1 (Syntax KBDIO$_{\text{CTL}}$). *Given a finite set A of agent names, a finite set G of group names, a finite set R of role names, a finite set C of role relations, and a countable set P of primitive proposition names, which includes '$g(a)$', '$r(a)$', and '$a\ ch\ b$' for all $a, b \in A$, $g \in G$, $r \in R$, and $ch \in C$, the admissible formulae of KBDIO$_{\text{CTL}}$ are recursively defined by:*

S1 Each primitive proposition in P is a state formula.
S2 If α and β are state formulae, then so are $\alpha \wedge \beta$ and $\neg\alpha$.
S3 If α is a path formula, $E\alpha$ and $A\alpha$ are state formulae.
S4 If α is a state formula and $a, b \in A$, then $K_a(\alpha), B_a(\alpha), D_a(\alpha), I_a(\alpha), O_{a,b}(\alpha)$ are state formulae as well.
P If α and β are state formulae, then $X\alpha$ and $\alpha U \beta$ are path formulae.

We assume the following abbreviations:

disjunction	$\alpha \vee \beta \equiv_{def} \neg(\neg\alpha \wedge \neg\beta)$	implication	$\alpha \rightarrow \beta \equiv_{def} \neg\alpha \vee \beta$
future	$F(\alpha) \equiv_{def} \top U \alpha$	globally	$G(\alpha) \equiv_{def} \neg F(\neg\alpha)$
permission	$P_{a,b}(\alpha) \equiv_{def} \neg O_{a,b}(\neg\alpha)$	prohibition	$F_{a,b}(\alpha) \equiv_{def} \neg P_{a,b}(\alpha)$
undirected	$O_a(\alpha) \equiv_{def} O_{a,a}(\alpha)$.		

The semantics of KBDIO$_{\text{CTL}}$ involves two dimensions. The truth of a formula is evaluated relative to a world w and a temporal state s. A pair $\langle w, s\rangle$ is called a situation. The relation between situations is traditionally called an accessibility relation (for beliefs) or a successor relation (for time).

Definition 2 (Situation structure KBDIO$_{\text{CTL}}$). *Assume a finite set A of agent names. A structure $M = \langle \Delta, \mathcal{R}, \mathcal{K}, \mathcal{B}, \mathcal{D}, \mathcal{I}, \mathcal{O}, L\rangle$ forms a situation structure if Δ is a set of situations, $\mathcal{R} \subseteq \Delta \times \Delta$ is a binary relation such that $w = w'$ whenever $\langle w, s\rangle \mathcal{R} \langle w', s'\rangle$, $Z(a) \subseteq \Delta \times \Delta$ for the functions $Z \in \{\mathcal{K}, \mathcal{B}, \mathcal{D}, \mathcal{I}\}$ and $a \in A$, and $\mathcal{O}(a, b) \subseteq \Delta \times \Delta$ with $a, b \in A$ are binary relations such that $s = s'$ whenever $\langle w, s\rangle Z(a) \langle w', s'\rangle$ or $\langle w, s\rangle \mathcal{O}(a, b) \langle w', s'\rangle$, and L an interpretation function that assigns a particular set of situations to each primitive proposition. $L(p)$ contains all those situations in which p holds.*

A speciality of CTL is that some formulae – called path formulae– are not interpreted relative to a particular situation. What is relevant here are full paths. The reference to M is omitted whenever it is understood. Note that $\alpha U \beta$ is true if α is true until the last moment before the first one in which β is true (alternative definitions are used in the literature too).

Definition 3 (Semantics KBDIO$_{\mathbf{CTL}}$). *Given a set A of agent names. A full path in situation structure M is a sequence $\chi = \delta_0, \delta_1, \delta_2, \ldots$ such that for every $i \geq 0$, δ_i is an element of Δ and $\delta_i \mathcal{R} \delta_{i+1}$, and if χ is finite with δ_n its final situation, then there is no situation δ_{n+1} in Δ such that $\delta_n \mathcal{R} \delta_{n+1}$. We say that a full path starts at δ iff $\delta_0 = \delta$. If $\chi = \delta_0, \delta_1, \delta_2, \ldots$ is a full path in M, then we denote δ_i by χ^i $(i \geq 0)$.*

Let M be a situation structure, δ a situation, χ a full path and $a, b \in A$ two agents. The semantic relation $\models$ for KBDIO$_{CTL}$ is then defined as follows:

S1 $\delta \models p$ *iff* $\delta \in L(p)$ *and* p *is a primitive proposition*
S2 $\delta \models \alpha \wedge \beta$ *iff* $\delta \models \alpha$ *and* $\delta \models \beta$
$\delta \models \neg\alpha$ *iff* $\delta \models \alpha$ *does not hold*
S3 $\delta \models E\alpha$ *iff for some full path* χ *in* M *starting at* δ, *we have* $\chi \models \alpha$
$\delta \models A\alpha$ *iff for each full path* χ *in* M *starting at* δ, *we have* $\chi \models \alpha$
S4 $\delta \models K_a(\alpha)$ *iff for every* $\delta' \in \Delta$ *such that* $\delta \mathcal{K}(a) \delta', \delta' \models \alpha$
$\delta \models B_a(\alpha)$ *iff for every* $\delta' \in \Delta$ *such that* $\delta \mathcal{B}(a) \delta', \delta' \models \alpha$
$\delta \models D_a(\alpha)$ *iff for every* $\delta' \in \Delta$ *such that* $\delta \mathcal{D}(a) \delta', \delta' \models \alpha$
$\delta \models I_a(\alpha)$ *iff for every* $\delta' \in \Delta$ *such that* $\delta \mathcal{I}(a) \delta', \delta' \models \alpha$
$\delta \models O_{a,b}(\alpha)$ *iff for every* $\delta' \in \Delta$ *such that* $\delta \mathcal{O}(a,b) \delta', \delta' \models \alpha$
P $\chi \models X\alpha$ *iff* $\chi^1 \models \alpha$
$\chi \models \alpha U \beta$ *iff there is an* $i \geq 0$ *such that* $\chi^i \models \beta$ *and for all* $j (0 \leq j < i), \chi^j \models \alpha$

Like Rao and Georgeff, we use standard interpretations of these operators. $O_{a,b}$ is interpreted as a standard deontic operator KD [30], B as KD45, K as S5, and D, I as KD modal logic operators. The properties discussed in this paper characterize the relation between mental attitudes of a single agent. Properties can always be expressed at two levels. First, we can express that *all* obligations of an agent towards an agent satisfy a property. In that case, the obligations are characterized by this property. Second, properties may hold for one particular obligation only. In that case we may say that this particular obligation satisfies the property, but it does not characterize the agent's obligations in general. In this paper, we follow the convention that properties expressed using α are axioms, and thus α can be substituted by any propositional formula.

However, it is important to notice that all properties expressed relative to a group or role, such as $r(a) \rightarrow K_a\alpha$, can only be expressed as formulas, not as an axiom. The reason is, roughly, a condition like $g(a)$ or $r(a)$ should not be substituted by another proposition. For example, if $r(a) \rightarrow K_a\alpha$ is an axiom, then so is $q \rightarrow K_a\alpha$. An alternative way to formalize organizational structure in Rao and Georgeff's logic is to index modal operators by groups and roles, and thus write the above property as an axiom $K_{g(a)}\alpha$. The reason we made this choice of formalizing organizational structure in propositions is that the expressive power of the alternative representation is limited. The loss of relativized axioms is considered to be less severe, as the status of interaction axioms in this logic is problematic anyway, as discussed in Section 9.

4 Specification of organizational structure

Organizational structure is typically specified in terms of roles and role relations. When agent $x \in A$ plays the role $p \in R$ of passenger, and has not paid before travel started, then he or she is obliged to pay a fine to the public transport company s. This can be specified by the following set of formulas, for all $x, y \in A$. Note that sanctions are modelled as obligations too, and that the violation condition is expressed using the until operator of CTL.

$$(p(x) \wedge (\neg \mathrm{paid}_x U \mathrm{travel}_x)) \rightarrow O_{x,s} \mathrm{fine}_x$$

If the public transport company $s \in A$ has delegated the power to collect fines to the ticket controller, $c \in R$, we get $(p(x) \wedge c(y) \wedge (\neg \mathrm{paid}_x U \mathrm{travel}_x)) \rightarrow O_{x,y} \mathrm{fine}_x$. Such so-called delegation relations can become complex and are not further discussed in this paper. See for example, [16, 2].

In general, obligations are created by interaction. For example, in an electronic market where agents are buying and selling goods, a confirmation to buy creates a obligation to pay for the buyer and an obligation of shipping the goods for the seller. Obligations may also be created by the way a social system is designed. A social system typically contains stable relationships between roles, which affect the obligations of the agents in those roles. In particular, obligations can be based on the known or believed mental attitudes of agents standing in a role relation. For example, the role relation *adopts*$\in C$ between agent $a \in A$ and agent $b \in A$ can be characterized by the following axiom, which says that agent a adopts all obligations of agent b towards some other agent $c \in A$. The following formula schema can be instantiated for all agents $a, b, c \in A$, and proposition letter q. Obviously we have an analogous property when we replace knowledge (K) by belief (B).

$$(a \textit{ adopts } b \wedge K_a O_{b,c} q) \rightarrow O_{a,c} q$$

We can further specify obligation adoption with additional formulas. For example, the formula schema $(r(a) \wedge r(b)) \rightarrow a \textit{ adopts } b$ specifies that agent a adopts the obligations of agent b when they play the same role r in the organization. In a similar way, $K_a D_b \alpha \rightarrow O_{a,b} \alpha$ characterizes that agent a adopts the known desires of agent b as its obligations. Take a client-server system for example. When the server s believes that its client c desires a piece of information, then we can specify that the server s is obliged to see to it that client c gets this information. The following axiom schema characterizes the *slave_of* $\in C$ or “your wish is my command” role relation, which says that the desires or intentions of master $m \in A$ become the obligations of slave $s \in A$.

$$(s \textit{ slave\_of } m \wedge K_s I_m q) \rightarrow O_{s,m} q$$

For example, reconsider the running example and assume that the passenger has not paid. Now we need a detection mechanism to make sure that the sanction is applied. A ticket controller has the institutional power to make a passenger without a ticket pay a fine. However, the controller does not have the power to make any passenger pay a fine. There must be a pretext. This can be specified as a restricted instance of the master-slave principle listed above.

$$(p(x) \wedge c(y) \wedge K_x K_y \neg \mathrm{have\_ticket}_x \wedge K_x I_y \mathrm{fine}_x) \ \rightarrow O_{x,y} \mathrm{fine}_x$$

For violation detection, we first still have to specify that not having a ticket counts as evidence of not having paid. How to formalize such constitutive norms is an open problem in deontic logic, see for example [24, 18, 3]. A very simple specification in our specification language is $(g(x) \wedge \neg K_x \mathrm{have\_ticket}_x) \rightarrow K_x(\neg \mathrm{paid}_x U \mathrm{travel}_x)$, for all member agents x of some suitable group $g \in G$.

We can further extend the logic with new group related concepts to specify requirements on groups of agents. For example, the first axiom schemata for $x, y \in A$ characterizes the property that all members of group g must know each other and they must be able to have the role relation ch that they can communicate with each other. This is called acquaintance among members of a group. Groups and roles can also be combined. For example, for any organization it is important that agents recognize the roles that other agents are enacting. In human society, uniforms, location (behind a desk) or badges are used to this purpose. A group $g \in G$ in which a role $r \in R$ of an agent $a \in A$ is known to all agents is called *transparent*.

$$(g(x) \wedge g(y)) \rightarrow (K_x g(y) \wedge (x\ ch\ y)) \qquad\qquad (g(a) \wedge g(b) \wedge r(a) \rightarrow K_b r(a))$$

Related to transparency of roles is the property of delegation transparency, which states that agents must know of other agents on behalf of whom they are acting. So if some agent a delegates a job to b, a's role as a principal must be known. Verifying delegation chains is particularly important for legal applications, because the principle remains legally accountable.

A promising issue in the specification of multiagent organizations is the definition of a set of patterns for groups, roles and role relations. Patterns have proven to be very useful in several areas of software engineering. For example, assume that we wish to define a pattern for the role relation *leader* $\in C$ as the property that the agent fulfilling the role is able to communicate with the group members and vice versa. Also, a group leader must be able to delegate tasks to the group members and persuade them to have certain beliefs. In addition, the obligations of members of a group are the obligations of the group leader (a failure to satisfy an obligation by a group member is a failure to satisfy the obligation of the group leader), and the members should be committed to the task delegated to them. The following schemata characterize such a group leader. Let $a, x \in A$, $g \in G$, $leader$ and $com \in C$ be role relations that represent 'leader of' and 'able to communicate', respectively.

$$\begin{array}{ll}
\multicolumn{2}{l}{a \textit{ leader } x \rightarrow} \\
\quad K_a(a\ com\ x) \wedge K_x(x\ com\ a) \wedge & \text{(ability to communicate)} \\
\quad D_a AFI_x q \rightarrow AFI_x q \wedge & \text{(task delegation)} \\
\quad I_a B_x q \rightarrow AXB_x q \wedge & \text{(persuading members)} \\
\quad O_{x,a} q \rightarrow O_{a,a} q \wedge & \text{(obligation inheritance)} \\
\quad I_x AFq \rightarrow A(I_x AFq \cup (B_a q \vee \neg B_a EFq))) & \text{(committed to delegated tasks)}
\end{array}$$

An interesting question for further research is how standard patterns used in business modelling or software engineering can be formalized in our specification language. In this paper we do not consider this question, but we return to our running example.

5 Formalizing the norm of the running example

The public transport norm can be phrased as follows: any agent in the role of passenger travelling by public transport, should have paid for the trip. We choose to describe this norm in terms of a so called 'deadline obligation': "if $x \in A$ is playing the role of passenger $p \in R$, then x is obliged towards society s to see to it that there is no history in which x does not pay until x travels".

$$p(x) \rightarrow O_{x,s} \neg E((\neg \mathrm{paid}_x \wedge \neg \mathrm{travel}_x) U \mathrm{travel}_x)$$

The concept of deadline obligation is rather complex, as several alternative definitions can be given [6]. The concept depends on the particular interpretation of the until operator. The formula states that the obligation applies to any agent in the role of passenger. However, this formula does not describe behavior. The following formula, without the obligation, does:

$$p(x) \rightarrow \neg E((\neg \mathrm{paid}_x \wedge \neg \mathrm{travel}_x) U \mathrm{travel}_x)$$

Our definition of deadline obligations is inspired by Rao and Gerogeff's formalizations of commitment strategies. The main axioms discussed in temporal extensions of BDI logic are realism properties and commitment strategies, in particular in $\mathrm{BDI}_{\mathrm{LTL}}$ by Cohen and Levesque [9] and in $\mathrm{BDI}_{\mathrm{CTL}}$ by Rao and Georgeff [22, 21].

Realism puts a constraint on desires, with respect to what the agent believes about the state of the world. Some examples of realism properties are $B_a\alpha \rightarrow D_a\alpha$, for 'overcommitted realism' as defined in [9], $D_a\alpha \rightarrow \neg B_a \neg \alpha$ for 'weak realism' as defined in [22, 21], and $D_aEF\alpha \rightarrow B_aEF\alpha$ for 'strong realism'.

Commitment strategies are constraints on the process of intention reconsideration: under what circumstances is it allowed to drop an intention? Examples of commitment strategies are $I_aAF\alpha \rightarrow A(I_aAF\alpha \; UB_a\alpha)$, for 'blind commitment', and the more interesting $I_aAF\alpha \rightarrow A(I_aAF\alpha \; U(B_a\alpha \vee \neg B_aEF\alpha))$, called 'single minded commitment'. Whereas realism properties are static, commitment strategies are dynamic in the sense that they specify the temporal evolution of intentions. In the remainder of this section we define static and dynamic properties that involve obligations.

Rao and Georgeff's commitment strategies are examples of interactions of motivational attitudes and time. Such interactions also occur for desires and obligations. Cohen and Levesque [9] distinguish 'achievement goals' and 'maintenance goals'. Their definition in $\mathrm{BDI}_{\mathrm{LTL}}$ can be adapted to $\mathrm{KBDIO}_{\mathrm{CTL}}$ as the definition of $O^A_{a,b}$ below on the left. Cohen and Levesque do not give a definition for maintenance goals, but they characterize the difference as follows: "Achievement goals are those the agent believes to be false; maintenance goals are those the agent already believes to be true". This suggests that we can give a formula $O^M_{a,b}\alpha$ to express a maintenance obligation: $O^M_{a,b}\alpha \equiv_{def} B_a\alpha \wedge O_{a,b}AF\alpha$. Alternatively, we could define a maintenance obligation by the restriction that the goal or obligation should be maintained all the time.

$$O^A_{a,b}\alpha \equiv_{def} B_a\neg\alpha \wedge O_{a,b}AF\alpha \qquad O^M_{a,b}\alpha \equiv_{def} B_a\alpha \wedge O_{a,b}AG\alpha$$

Another issue are the conditions that may discharge an obligation. Obligations typically persist until a deadline, e.g., deliver the goods before noon, or they persist forever, e.g., don't kill. We denote a deadline obligation by $O_{a,b}(\alpha, d)$, where achievement of the proposition d is the deadline for the obligation to achieve α. A deadline obligation $O_{a,b}(\alpha, d)$ persists until it is fulfilled or becomes obsolete because the deadline is reached.

$$O_{a,b}(\alpha, d) \equiv_{def} A((O_{a,b}\alpha)U(\alpha \vee d))$$

A deadline obligation $O_{a,b}(\alpha, \alpha)$, for which the only deadline is the achievement of the obligation itself, is called a 'dischargeable obligation'. The definition simplifies to $O_{a,b}(\alpha, \alpha) \equiv_{def} A((O_{a,b}\alpha)U\alpha)$. Alternatively, we may characterize the property that obligations from agent a to agent b are dischargeable by the axiom $O_{a,b}\alpha \leftrightarrow A((O_{a,b}\alpha)U\alpha)$. Analogously we can also define dischargeable desires. For example, an agent may desire a receipt until it gets one. However, a drawback of the axiom is that it is expressed in terms of facts, which are not accessible to agents. We therefore replace the occurrence of α without a preceding modal operator by $K_a\alpha$. Moreover, again we believe that dischargeable obligations and dischargeable desires obey different discharging conditions. An obligation can only be discharged by the *knowledge* that the obliged condition is fulfilled. A desire can already be discharged by the *belief* that this is the case. Consequently, the property that obligations from agent a towards agent b are dischargeable, and analogously the property that desires from agent a are dischargeable, are characterized by the following two axioms, respectively.

$$O_{a,b}\alpha \leftrightarrow A((O_{a,b}\alpha)UK_a\alpha) \qquad D_a\alpha \leftrightarrow A((D_a\alpha)UB_a\alpha)$$

We can characterize that $O_{a,b}\alpha$ persists forever, i.e., that it is a 'non-dischargeable obligation', by $O_{a,b}\alpha \leftrightarrow AGO_{a,b}\alpha$. We can also combine the definitions, such that agents for instance have non-dischargeable achievement obligations, or dischargeable maintenance obligations.

As we now have specified the norm, we finally specify the four ways to realize that the norm is fulfilled. First we regiment the norm into the environment, such that agents cannot violate the norm. Then we define agents which are designed such that they cannot violate norms. Finally we discuss formalizations that rely on rationality or social control. In the formalization, we distinguish between assumptions about societies, ticket policies, individual agents and the environment. These assumptions are either formalized as formulas or as axioms. The difference is roughly that axioms are true in any world of the model, and for axioms we can substitute the propositions by other propositions. The norm itself – the first formula above – can be part of those assumptions. We want to verify whether the second property follows from this. $\mathit{\Gamma}_{ins}$ is a set of formulas representing assumed properties of the institution, in this case the public transport network, $\mathit{\Gamma}_{r_1}, ..., \mathit{\Gamma}_{r_n}$ are sets of formulas that represent the assumed properties for the various roles $r_1, ..., r_n$ in the institution, like passenger or ticket collector, $\mathit{\Gamma}_{env}$ is a set of formulas representing the assumed properties of the behavior of the environment, and $\mathit{\Delta}$ represents the property to be shown. As usual we use the weakest version of modal entailment, i.e., $\varphi \models \psi$ holds if and only if it is the case that when φ is satisfied in some state of a model, than also ψ is satisfied.

6 Implementing a norm in the environment

An important question when developing a normative system is whether the norms can be violated or not, i.e., whether the norms are soft or hard constraints. In the latter case, the norms are said to be regimented. Regimented norms correspond to preventative control systems in computer security [15]. For example, in the metro example it is not possible to travel without a ticket, because there is a preventative control system, whereas it is possible to travel without a ticket on the French trains, because there is a detective control system. Norm regimentation for agent a is characterized by the following axiom.

$$O_{a,b}\alpha \to \alpha$$

The following example illustrates the specification of regimentation in the running example. It also illustrates that regimentation can be specified at different levels of abstraction. At the detailed level, it is specified precisely how the norm is implemented in the environment. At a more abstract level, the norm is given as an axiom, and it is specified that the norm is regimented - but not *how* it is regimented.

Example 1 (norm enforcement by imposing a restrictive environment). The set of agents is $A = \{x, s\}$, the set of roles is $R = \{p\}$, the set of groups and role relations is $G = ch = \emptyset$, and the set of propositions is $P = \{\text{travel}_x, \text{have\_ticket}_x, \text{paid}_x, \text{climbed\_barrier}_x, \text{pass\_barrier}_x\}$. The following formulas represent assumptions. (1) Having a ticket is the evidence for having paid. (2) Passengers cannot climb the barrier. (3) To travel, a passenger must have passed the barrier. (4) To pass the barrier, a passenger must have paid, or must have climbed it.

$$\begin{array}{lll}
1.\ \Gamma_{\text{ins}} & = & \{p(x) \to AG((\text{have\_ticket}_x \to \text{paid}_x))\}, \\
2.\ \Gamma_{\text{passenger}} & = & \{AG(\neg\text{climb\_barrier}_x)\}, \\
3.\ C & = & \{\neg E(\neg\text{pass\_barrier}_x U \text{travel}_x), \\
4. & & \quad AG(\text{pass\_barrier}_x \leftrightarrow (\text{have\_ticket}_x \vee \text{ climb\_barrier}_x))\}
\end{array}$$

We now show that $p(x) \to \neg E((\neg\text{paid}_x \wedge \neg\text{travel}_x))U\text{travel}_x)$ follows from the above set. Suppose no passenger is travelling; then the behavior is trivially satisfied. Now suppose a passenger is travelling. That means that she passed the barrier (3). That means she has a ticket, or else she climbed the barrier (4). This last option is ruled out by assumption (2). So she has a ticket, which means she paid (1).

Instead, we can specify the system at a higher level of abstraction by specifying the norm and specifying that the norm is regimented.

$$\begin{array}{lll}
1.\ \Gamma_{\text{ins}} & = & \{p(x) \to O_{x,s}\neg E((\neg\text{paid}_x \wedge \neg\text{travel}_x)U\text{travel}_x)\}, \\
2.\ \Gamma_{\text{passenger}} & = & \{O_{x,s}\alpha \to \alpha\}, \\
3.\ \Gamma_{\text{env}} & = & \{\}
\end{array}$$

Note that the first formula is an ordinary assumption, whereas the second formula is an axiom of the logic. The desired consequence $p(x) \to \neg E((\neg\text{paid}_x \wedge \neg\text{travel}_x)U\text{travel}_x)$ follows directly from the assumptions.

7 Implementing a norm by designing norm abiding agents

A drawback of the regimentation property in the previous section is that it is not expressed in terms of mental concepts, and thus agents cannot reason about it. Therefore we strengthen it to the case in which not only α is the case, but the agent also knows that this is the case. The property that the obligations of agent a towards agent b are regimented is characterized by the following axiom.

$$O_{a,b}\alpha \rightarrow K_a\alpha$$

Note that since we have the axiom $K_a\alpha \rightarrow \alpha$, we have that $O_{a,b}\alpha \rightarrow K_a\alpha$ implies $O_{a,b}\alpha \rightarrow \alpha$. This strong property can be weakened in various directions. First, we can weaken it in the sense that it is not necessarily a fact that the obligation is obeyed, but that at least the opposite is not the case, $O_{a,b}\alpha \rightarrow \neg K_a\neg\alpha$. Second, it can be weakened such that agents *believe* that the obligation is not violated: $O_{a,b}\alpha \rightarrow B_a\alpha$ and $O_{a,b}\alpha \rightarrow \neg B_a\neg\alpha$. Third, the time of compliance to the obligation can be weakened: $O_{a,b}\alpha \rightarrow K_aAF\alpha$, or e.g., $O_{a,b}\alpha \rightarrow K_aAX\alpha$, etc.

At the most abstract level, the formalization of the running example remains nearly the same, we replace the regimentation axiom by the epistemic variant above. Moreover, the logic can specify the decision making of agents at more detailed levels. In particular, the logic can specify when desires or obligations lead to intentions, and when intentions lead to actions. That is, the regimentation axiom $O_{a,b}\alpha \rightarrow K_a\alpha$ is decomposed into the following two axioms.

$$O_{a,b}\alpha \rightarrow I_a\alpha \qquad I_a\alpha \rightarrow K_a\alpha$$

Furthermore, there are many variants on these two axioms. For example, a variant of regimentation concerns conditionality with respect to a conflict between an agent's internal and external motivations. For example, 'if an agent is obliged to buy a ticket, but desires to spend no money, then he intends to buy the ticket anyway, because he is a 'social' agent that does not let his own desires overrule his obligations'. The property that agent a is strongly or weakly respectful with respect to agent b is characterized by the following two axioms. The second formula is implied by the first one if the D axiom $\neg(I_a\alpha \wedge I_a\neg\alpha)$ holds for modality I_a.

$$(O_{a,b}\alpha \wedge D_a\neg\alpha) \rightarrow I_a\alpha \qquad (O_{a,b}\alpha \wedge D_a\neg\alpha) \rightarrow \neg I_a\neg\alpha$$

Finally, the intention of achieving a state can interact with obligations to satisfy the conditions for achieving that state. In such a case, new intentions are implied. The interaction between intention and norms and the creation of intentions can be formulated as the following benevolent axiom:

$$I_x\alpha \wedge O_{x,s}\neg E(\neg\beta U\alpha) \rightarrow I_x\beta$$

The specification of rational agents is one of the main issues studied in agent theory, and these results can be reused in KBDIO$_{\mathrm{CTL}}$. However, it is also well known that modal logic has to be extended in several ways to make detailed agent models. For example, to specify agents that maximize expected utility BDI$_{\mathrm{CTL}}$ has to be extended in various ways [20].

8 Implementing a norm by relying on rationality or social control

The first way in which norms can be implemented, is to rely on agent rationality and impose fines on norm violations. As mentioned above, the logic can specify when desires or obligations lead to intentions, and when intentions lead to actions. In particular, in the previous section the regimentation axiom $O_{a,b}\alpha \rightarrow K_a\alpha$ is decomposed into $O_{a,b}\alpha \rightarrow I_a\alpha$ and $I_a\alpha \rightarrow K_a\alpha$. In this section, we make sure that the agent *desires* to fulfill the obligation. That is, the regimentation axiom $O_{a,b}\alpha \rightarrow K_a\alpha$ is decomposed into the following three axioms.

$$O_{a,b}\alpha \rightarrow D_a\alpha \qquad D_{a,b}\alpha \rightarrow I_a\alpha \qquad I_a\alpha \rightarrow K_a\alpha$$

We thus interpret the first axiom as the specification that the system is such that it is desired to fulfill the obligation. However, there are several ways in which the axiom can be interpreted. The first explanation is that the agent is norm abiding and *internalizes* its obligations in the sense that they turn into desires. For example, if an agent is obliged to buy a ticket, then it also desires to buy a ticket. The axiom can be weakened to the condition that at least the agent cannot decide to violate the obligation, e.g., at least it cannot desire not to buy a ticket: $O_{a,b}\alpha \rightarrow \neg D_a\neg\alpha$. Instead of respectful, agents may also be egocentric, which can be characterized by similar properties like $(O_{a,b}\alpha \wedge D_a\neg\alpha) \rightarrow I_a\neg\alpha$ and $(O_{a,b}\alpha \wedge D_a\neg\alpha) \rightarrow \neg I_a\alpha$.

The second interpretation of $O_{a,b}\alpha \rightarrow D_a\alpha$ is that the obligation turns into a desire, because violating the desire implies a fine. We already discussed fines in Section 4. The following example is a simplified version, that illustrates how the desire not to be fined can lead to the desire to fulfill obligations. desires. Note that in this formalization the derived desire may also be interpreted as a goal, which is often the case in BDI$_{\text{CTL}}$ specifications.

$$O_{a,b}\text{paid} \rightarrow K_a(\neg\text{paid} \rightarrow \text{fine}) \qquad (K_a(\neg\text{paid} \rightarrow \text{fine}) \wedge D_a\neg\text{fine}) \rightarrow D_a\text{paid}$$

The third interpretation of $O_{a,b}\alpha \rightarrow D_a\alpha$ is that violating the obligation leads to social embarrassment. This can be specified analogously to fines.

9 Related work

Despite the popularity of Roa and Georgeff's logic in agent theory to specify and verify multiagent systems, the logical analysis of their logic is still in its infancy. Rao and Georgeff did not present a full axiomatization of their logic, which was only presented much more recently by Schild's reduction to the μ calculus. Moreover, the axiomatization is restricted to the logic without any interaction axioms. In the meantime, logicians have restricted themselves to small fragments of their logic, for example to study the interaction between knowledge and time, or to study the interaction between beliefs and obligations.

Within deontic logic in computer science, our work is most closely related to dynamic deontic logic, extensions of dynamic logic with modalities for obligations and permissions. In multiagent systems, recently norms and normative systems are discussed, but their specification or verification has has not been addressed. In action programs in IMPACT, there is a discussion on whether obligations can be violated, i.e., on norm regimentation [26]. We have addressed this issue in the context of the BOID project, see http://boid.info. The present paper extends our short paper [5].

10 Summary

The motivation of our work is how such normative computer systems can be specified. This problem breaks down as follows:

1. How to develop a logic for specification of normative computer systems?
2. Which kind of properties can be expressed in the specification logic?
3. How to apply the specification logic to application domains?

Our methodology is to specify properties involving obligations in an extension of Rao and Georgeff's $\mathrm{BDI}_{\mathrm{CTL}}$ [21, 23, 11, 12]. Such an extension consists of an extension of the logic and an extension of the properties expressed in the logic. Obligations are motivational attitudes, just like desires, but they also have organizational aspects. This can be represented, for example, by introducing roles and by formalizing obligation as a directed modality. Thus, whereas we may say that agent a desires to prepare a report, we say that the agent a is obliged to prepare a report *towards another agent* b. We accomplish our extension of $\mathrm{BDI}_{\mathrm{CTL}}$ with obligations in the following steps:

- The introduction of an extension of $\mathrm{BDI}_{\mathrm{CTL}}$ called $\mathrm{KBDIO}_{\mathrm{CTL}}$, that makes the distinction between desires and obligations explicit, as well as the distinction between beliefs and knowledge. We extend $\mathrm{BDI}_{\mathrm{CTL}}$ with directed obligations [17, 13, 25, 4, 27] and roles.
- We introduce various single agent and multiagent properties. These properties can be used in a high-level design language for normative computer systems.
- We apply the logic and the properties to the implementation of an organizational norm.

References

1. F. Arbab, F.S. de Boer, M.M. Bonsangue, and J.V. Guillen Scholten. A channel-based coordination model for components. Technical Report SEN-R0127, CWI, Amsterdam, 2001.
2. O.L. Bandmann, B. Sadigi Firozabadi, and M. Dam. Constrained delegation. In *Procs. of IEEE Symposium on Security and Privacy 2002*, pages 131–140, 2002.
3. G. Boella and L. van der Torre. Regulative and constitutive norms in normative multiagent systems. In *Procs. KR'04*, Whistler, CA, 2004.
4. J. Broersen, M. Dastani, Z. Huang, and L. van der Torre. Trust and commitment in dynamic logic. In *Procs. of EurAsia ICT 2002*, LNCS 2510, pages 677–684. Springer, 2002.
5. J. Broersen, M. Dastani, and L. van der Torre. BDIO_CTL: Properties of obligation in agent specification languages. In *Procs. of IJCAI'03*, pages 1389–1390, 2003.
6. J. Broersen, F. Dignum, V. Dignum, and J.-J. Meyer. Designing a deontic logic of deadlines. In *Procs. of DEON'04*, LNCS. Springer, 2004. This volume.

7. J. Carmo and O. Pacheco. A role based model for the normative specificatiuon of organized collective agency and agents interaction. *Autonomous Agents and Multi-Agent Systems*, 6:145–184, 2003.
8. C. Castelfranchi. Modelling social actions for AI agents. *Artificial Intelligence*, 103:157–182, 1998.
9. P.R. Cohen and H.J. Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42(3):213–261, 1990.
10. M. Dastani, V. Dignum, and F. Dignum. Role assignment in open agent societies. In *Procs. of AAMAS'03*, pages 489–496. ACM, 2003.
11. M. Dastani and L. van der Torre. An extension of BDICTL with functional dependencies and components. In *Procs. of LPAR'02*, LNCS 2514, pages 115–129. Springer, 2002.
12. M. Dastani and L. van der Torre. Specifying the merging of desires into goals in the context of beliefs. In *Procs. of EurAsia ICT 2002*, LNCS 2510, pages 824–831. Springer, 2002.
13. F. Dignum. Autonomous agents with norms. *Artificial Intelligence and Law*, 7(1):69–79, 1999.
14. J. Ferber and O. Gutknecht. A meta-model for the analysis and design of organizations in multi-agent systems. In *Procs. of ICMAS'98*, pages 128–135. IEEE Press, 1998.
15. B. Sadagi Firozabadi and L. van der Torre. Towards an analysis of control systems. In *Procs. of ECAI'98*, pages 317–318, 1998.
16. B. Sadigi Firozabadi and M. J. Sergot. Revocation schemes for delegated authorities. In *Procs. of POLICY'02*, pages 210–213, 2002.
17. H. Herrestad and C. Krogh. Obligations directed from bearers to counterparties. In *Procs of ICAIL'95*, pages 210 – 218, New York, 1995.
18. A. Jones and M. Sergot. A formal characterisation of institutionalised power. *Journal of IGPL*, 3:427–443, 1996.
19. J. Meyer and R. Wieringa. *Deontic Logic in Computer Science: Normative System Specification*. John Wiley and Sons, 1993.
20. A. S. Rao and M. P. Georgeff. Deliberation and its role in the formation of intentions. In *Procs. of UAI-91*, 1991.
21. A. S. Rao and M. P. Georgeff. Decision procedures for BDI logics. *Journal of Logic and Computation*, 8(3):293–343, 1998.
22. A.S. Rao and M.P. Georgeff. Modeling rational agents within a BDI-architecture. In J. Allen, R. Fikes, and E. Sandewall, editors, *Procs. of KR'91*, pages 473–484. Morgan Kaufmann Publishers, 1991.
23. K. Schild. On the relationship between BDI-logics and standard logics of concurrency. *Autonomous agents and multi-agent systems*, 3:259–283, 2000.
24. J.R. Searle. *The Construction of Social Reality*. The Free Press, New York, 1995.
25. M. P. Singh. An ontology for commitments in multiagent systems: toward a unification of normative concepts. *Artificial Intelligence and Law*, 7:97–113, 1999.
26. G. Pick T. Eiter, V.S. Subrahmanian. Heterogeneous active agents, I: Semantics. *Artificial Intelligence*, 108:179–255, 1999.
27. Y. Tan and W. Thoen. Modeling directed obligations and permissions in trade contracts. In *Procs of HICCS'98*, pages 166–175, 1998.
28. R. Wieringa and J. Meyer. Applications of deontic logic in computer science: A concise overview. In *Deontic Logic in Computer Science*, pages 17–40. John Wiley & Sons, Chichester, England, 1993.
29. M.J. Wooldridge, N.R. Jennings, and D. Kinny. The Gaia methodology for agent-oriented analysis and design. *Autonomous Agents and Multi-Agent Systems*, 3(3):285–312, 2000.
30. G.H. von Wright. Deontic logic. *Mind*, 60:1–15, 1951.