

# Combining Norms, Roles, Dependence and Argumentation in Agreement Technologies

Patrice Caire<sup>\*a</sup>

Leendert van der Torre<sup>ab</sup>

Serena Villata<sup>c</sup>

<sup>a</sup>*Interdisciplinary Center for Security, Reliability and Trust, University of Luxembourg*

<sup>b</sup>*Computer Science and Communication, University of Luxembourg*

<sup>c</sup>*INRIA Sophia Antipolis, France*

## Abstract

A major challenge for Agreement Technologies is the combination of existing technologies and reasoning methods. In this paper we focus on the three core layers of the Agreement Technologies tower, called Norms, Organization and Argumentation. We present a framework for arguing about agreements based on norms, roles and dependence, together with a case study from the sharing economy.

## 1 Introduction

Agreement technologies refer to computer systems in which autonomous software agents negotiate with one another, typically on behalf of humans, to come to mutually acceptable agreements.<sup>1</sup> The concept “agreement” bridges individual and collective reasoning, i.e. the micro and macro level studied in economics and the social sciences. The agreement technologies handbook [10] distinguishes five technologies: for semantic alignment and interoperability, for normative and legal reasoning, for organizations, for argumentation and negotiation, and for trust and reputation management. They are developed in distinct research communities, and this plurality of methods is seen as a strength rather than a weakness. Consequently, a major challenge is how to *combine* existing technologies and reasoning methods. This leads to our research problem:

**Question.** How to *combine* norms, roles, dependence and argumentation for Agreement Technologies?

Boella and van der Torre [3] sketch an architecture combining reasoning methods for all five layers using abstraction, Billhardt *et al.* [2] propose to represent agreements using dependence, and in earlier work [5] we propose an abstract model for arguing about agreements using dependence networks. Building on these ideas, we introduce an abstract model for arguing about agreements with organizational roles and norms.

In our new model, we represent a possible agreement as a role based social dependence network, that is, a labeled graph on role playing agents. The agents in a dependence are called the depender and performer respectively, and the labels are reasons (goals, tasks, resources and norms), together with cost and benefit functions for the performer and the depender respectively. The goals, tasks and resources are associated with the agents, whereas the norms are associated with the roles the agents play. We consider also agreements about which norms are in force, and agreements on which roles the agents play.

We do *not* study normative reasoning itself, or dependence networks, role assignment, or argumentation—each of which has a huge literature [10]—but we consider only their interaction. Moreover, we consider neither the lowest layer of semantic alignment and interoperability, nor the upper layer of trust and reputation.

The layout of this paper is as follows: Section 2 introduces the problem of combining reasoning and how it has been addressed in the literature. Section 3 and section 4 describe the kind of agreements we can reason on, and how roles and norms influence such agreements using a real world scenario of sharing economy.

---

<sup>\*</sup>The present research is supported by the National Research Fund, Luxembourg, CoPAInS project (code: CO11/IS/1239572).

<sup>1</sup>See the website of the European Network <http://www.agreement-technologies.eu/> for a historical development of agreement technologies.

## 2 Methodology

### 2.1 Combining reasoning

We use logic and reasoning as our methodology, and we are inspired by two authors. First, Dov Gabbay [9] combines logics using labeled deductive systems and fibring semantics, and more recently his network perspective combines or integrates logical methods with, for example, neural network reasoning and Bayesian network reasoning. He promotes formal approaches to practical (individual) reasoning, and more recently he studies normative reasoning and argumentation as ways to combining individual and collective reasoning.

Second, Johan van Benthem [13] uses recursive axiomatizations in modal logic as a general approach to combine logics of individual agents, distinguishing dynamic epistemic logics, preference logics, logics of questions, game logics, and more. An application of this theory is in understanding natural language, where combining modalities and conditionals is a central challenge.

We use individual and collective reasoning methods developed in artificial intelligence and multiagent systems. For example, knowledge based systems typically combine various reasoning methods. As another example, IBM's Watson<sup>2</sup> integrates specialized reasoners for space and time in its system. Our work in AI is in line with the work in philosophy and logic of Gabbay and van Benthem, and that there can be fruitful exchange of ideas and methodologies between philosophy, logic and artificial intelligence.

### 2.2 Dependence networks

Castelfranchi and colleagues [7, 12, 4] develop a theory of dependence networks, applicable to most social interactions. The abstract dependence networks we use in this paper are directed graphs labeled with various kinds of reasons [15]: the fulfillments of *goals* and ought-to-be obligations, the execution of *tasks* and ought-to-do obligations, and the production of *resources*. Our model of arguing about dependence [5] follows Sauro [11] in that every reason can occur only in the dependence relations of a single depender. In contrast to Sauro, we represent only OR-dependence, no AND-dependence, such that we do not consider jointly performed tasks, agents together fulfilling a goal, or together providing a resource. In other words, we assume that tasks, goals and resources can be performed, fulfilled or provided by a single agent as in [6].

For the individual agents, our model of arguing about dependence [5] uses a standard cost-benefit analysis. Each dependence comes with a cost for the performer, and a benefit for the depender, which are positive real numbers. The payoff for an agent is the sum of the benefits of each reason where he depends, minus the sum of the costs of the reasons where he performs. An investigation into uncertainty, utility, and the dependence among reasons, are left for future research.

### 2.3 Case study: the sharing economy

Agreement technologies are developed around a number of case studies, and we believe that such case studies are indispensable. An example of an extensive case study in combining reasoning is work of Gabbay and colleagues on formalizing the Talmud [1]. We are developing a case study on the sharing economy.

**Example 1 (Sharing economy)** *In the collaborative or sharing economy, owners of underused resources connect with others willing to pay to use them. The connection between owners and users may be made directly or peer-to-peer, or indirectly via a service. For example, car sharing has environmental benefits, reduces the cost of fuel and tolls, and allows commuters who are more than two per car to use the carpooling lanes on highways during peak traffic hours and thereby, to save time.*

*Our running example is a peer-to-peer car sharing with on the one hand, two car owners, Betty and Cathy, each driving their own car and looking to give a ride to a single passenger, for example to share the costs and save money. On the other hand, two other people, Arthur and Dana, both need to get a ride, Arthur to visit his mother, and Dana to go to work. Furthermore, Arthur prefers to get a ride with Betty over sharing the car of Cathy. Betty does not want to drive with him, but she is obliged to do so.*

---

<sup>2</sup><http://www-03.ibm.com/innovation/us/watson/>

### 3 Role based agreements

#### 3.1 Role playing agent dependence networks

An abstract dependence network is a labeled graph, represented by a binary relation  $D$  among abstract nodes represented by  $N$ , and labeled by  $E$ . If  $(d, p, e) \in D$  for  $d, p \in N$  and  $e \in E$ , then we say that  $d$  depends on  $p$  for reason  $e$ . Following the usual terminology in this area, we also say that  $p$  has power over  $d$  due to reason  $e$ . We call node  $d$  the depender, and node  $p$  the performer of the dependence. Each dependence comes with a real valued benefit for the depender, and a real valued cost for the performer.

**Definition 1 (Abstract dependence network)** A dependence network  $DN$  is a tuple  $\langle N, E, D, c, b \rangle$  where  $N$  and  $E$  are two disjoint sets, and  $D \subseteq N \times N \times E$  is a binary (dependence) relation over  $N$  for each element of  $E$ , and  $c, b : N \times E \rightarrow \mathbb{R}^0$  are (cost, benefit) functions associating non-negative real numbers with each pair of elements of  $N$  and  $E$ .

We distinguish three kinds of abstract dependence networks, namely agent dependence networks, role dependence networks, and role playing agent dependence networks.  $N$  is instantiated with agents, roles or role playing agents respectively, and  $E$  is instantiated with goals, tasks, resources (reasons for agent dependence), or norms (reasons for role dependence). The role playing agent dependence network is defined in terms of the other two, together with a relation  $plays$ , which are pairs of an agent and a role, representing the role playing agents. We make several simplifying assumptions to facilitate the formal framework. Labels are specific to dependers, in the sense that distinct dependers cannot depend for the same reason. Agent reasons and role reasons are distinct sets, and every role is played by at most one agent. We represent only OR-dependence, no AND-dependence, i.e., we do not consider jointly performed tasks, agents together fulfilling a goal, or together providing a resource. In other words, we assume that tasks, goals and resources can be performed, fulfilled and provided by a single agent. These constraints are not realistic for some applications, and we are aware that they have to be relaxed in future work.

**Definition 2 (Role playing agent dependence network)** Agent dependence networks  $ADN = \langle A, E_1, D_1, c_1, b_1 \rangle$  and role dependence networks  $RDN = \langle R, E_2, D_2, c_2, b_2 \rangle$  are abstract dependent networks such that  $(d_1, p_1, e), (d_2, p_2, e) \in D_1$  or  $D_2$  implies  $d_1 = d_2$ . An agent role assignment  $plays \subseteq A \times R$  is a set of pairs of agents and roles, such that  $(a_1, r), (a_2, r) \in plays$  implies  $a_1 = a_2$ . If  $A \cap R = \emptyset$  and  $E_1 \cap E_2 = \emptyset$ , then the role playing agent dependence network for  $ADN$ ,  $RDN$  and  $plays$  is an abstract dependence network  $\langle plays, E, D, c, b \rangle$  such that  $E((a, r)) = E_1(a) \cup E_2(r)$ ,  $D((a_1, r_1), (a_2, r_2), e)$  if and only if  $D_1(a_1, a_2, e)$  or  $D_2(r_1, r_2, e)$ ,  $c((a, r), e) = c_1(a, e)$  if  $e \in E_1$ ,  $c_2(r, e)$  if  $e \in E_2$ , and  $b((a, r), x) = b_1(a, x)$  if  $e \in E_1$ ,  $b_2(r, x)$  if  $e \in E_2$ .

Example 2 illustrates how three possible agreements for sharing a car are built up.

**Example 2 (continued)** Consider the dependence network visualized in Figure 1.1. By convention, a dependence of  $d$  on  $p$  for the reason  $e$  is visualized by an arrow from  $d$  to  $p$  labeled with  $e$ . For example, if the reasons represent resources, then the flow of resources is inverse to the direction of the arrows. To read these figures, it is easier to start from Figure 1.4 on the right hand side, and see how it is built up using the other three figures. In Figure 1.4, three large ovals are visualized one above the other, to represent the three proposals for agreements: Arthur shares Betty's car, Arthur shares Cathy's car, or Dana shares Cathy's car. Figure 1.4 is explained in more detail in Example 3 and 4. A dependence in Figure 1.4 can originate from the agents in Figure 1.1, or from the roles they play in Figure 1.2, and the reasons, costs and benefits visualized in the figures explain in more detail why a role playing agent depends on another role playing agent. The set of role playing agents is given by the agent role assignment in Figure 1.3, but  $a : r_1$  and  $c : r_3$  are visualized twice in Figure 1.4 to make the proposals for agreements more explicit. Every dependence in Figure 1.4 is derived from either the agent dependence network in Figure 1.1, or the role dependence network in Figure 1.2. In Figure 1.1, the four nodes represent the agents  $a$  for Arthur,  $b$  for Betty,  $c$  for Cathy and  $d$  for Dana. Dependence  $(a, b, e_1)$  may be read as "agent  $a$ , Arthur, depends on agent  $b$ , Betty, to visit

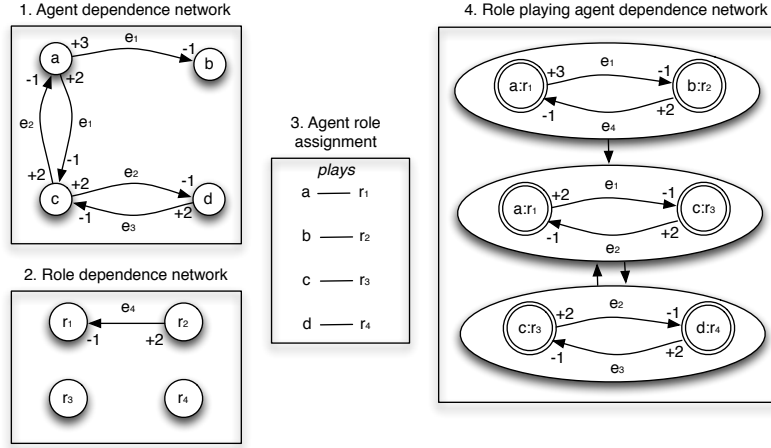


Figure 1: Building agreements: Agent dependence (1), Role dependence (2), Role assignment (3) and the combination (4).

his mother,”  $(a, c, e_1)$  as “agent a, Arthur, depends on agent c, Cathy, to visit his mother,” and  $(c, a, e_2)$  as “agent c, Cathy, depends on agent a, Arthur, to save money.” The reason  $e_1$  can be a goal of the agent a to visit his mother, or a task of Betty or Cathy to give a ride for the visit, or the car shared for the visit can be a resource. The benefit for a to visit his mother by riding with b (3), is greater than by riding c (2), whereas the cost for b and c is 1. We write costs with a “-” sign and benefits with a “+”. Furthermore, dependence  $(d, c, e_3)$  may be read as “agent d, Dana, depends on agent c, Cathy, to go to work,  $e_3$ ,”  $(c, d, e_2)$  as “Cathy depends on Dana to save money,  $e_2$ ,” and  $(c, a, e_2)$  as “Cathy, depends on Arthur to save money,  $e_2$ ”.

Figure 1.2, visualizes a role network. A norm “carpooling lane can only be used if you share, i.e., if you are two or more people in the car” creates a dependency. In our example, the graph contains only the actual norm  $e_4$  between roles  $r_2$  and  $r_1$ . Normative dependence  $(r_2, r_1, e_4)$  may be read as “driver 1 depends on passenger 1 to be able to use the carpooling lane.” We now consider the agent role assignment visualized in Figure 1.3. Role assignment  $(a, r_1)$ , may be read as “agent a, Arthur, plays the role  $r_1$ , passenger 1,”  $(b : r_2)$  as “agent b, Betty, plays the role  $r_2$ , driver 1,”  $(c, r_3)$ , may be read as “agent c, Cathy, plays the role  $r_3$ , driver 2,”  $(d : r_4)$  as “agent d, Dana, plays the role  $r_4$ , passenger 2.” Figure 1.4 visualizes the combined models. Each one of the three possible situations is depicted within an oval. Each double-circle node depicts an agent/role, for example  $(a : r_1)$  can be read as “agent a, Arthur, plays the role of passenger 1.” We have multiple instances of drivers and passengers because each role can be played by one agent only (see Sauro [11] for a further explanation and discussion, they are sometimes called role instances).

### 3.2 Proposals and agreements

We associate sets of proposals for agreements with dependence networks. A *proposal* for an agreement is a dependence relation, and a *proposal function* is a function from abstract dependence networks to sets of proposals. Dependence  $(d, p, e)$  proposes a commitment of the performer  $p$  to act in the benefit of the depender  $d$  to fulfill a goal, perform a task, or provide a resource.

**Example 3 (continued)** The three proposals for agreements are visualized in Figure 1.4. by the three ovals, and each oval is a dependence networks itself. For each role playing dependence network, the proposal function gives all these proposals.

For modeling the interaction among the agents, we use a standard definition from game theory [14]. A proposal  $P_1$  dominates a proposal  $P_2$  if and only if for all agents involved in  $P_1$ , the pay-off in  $P_1$  is at least

as good as in  $P_2$ , and for at least one of them it is strictly better. The restriction to agents of  $P_1$  is crucial: there may be agents who are worse off in  $P_2$ , but they cannot argue against it. For the costs and benefits of the agents, we add the costs and benefits for all the roles they play. It may be that an agent has a negative pay-off for one of its roles, as long as the sum of all payoffs for all the roles he is playing is positive.

**Definition 3 (Undominated proposal)** Let  $DN = \langle play, E, D, c, b \rangle$  be a role playing dependence network. A proposal  $P \subseteq D$  for  $DN$  is a dependence relation. A proposal function  $p$  is a function from role playing dependence networks to sets of proposals  $p(\langle play, R, D, c, b \rangle) \subseteq 2^D$ . The reasons where agent  $a$  has a benefit in proposal  $P$ , written as  $benefits(a, P)$ , are  $\{e \in E \mid p \in A, r_1, r_2 \in R, ((a, r_1), (p, r_2), e) \in P\}$ . Analogously, the reasons where he has a cost in proposal  $P$ , written as  $costs(a, P)$ , are  $\{e \in E \mid d \in A, r_1, r_2 \in R, ((d, r_1), (a, r_2), e) \in P\}$ . The payoff of a proposal  $P$  for agent  $a$ , written as  $pay-off(a, P)$ , is  $\sum_{e \in benefits(a, P)} b(a, e) - \sum_{e \in costs(a, P)} c(a, e)$ . We write  $A(P) = \{a \in A : \exists((a, r_1), (p, r_2), e) \in P \text{ or } \exists((d, r_1), (a, r_2), e) \in P\}$  for the agents of a proposal  $P$ . A proposal  $P_1$  dominates proposal  $P_2$ , written as  $P_1 > P_2$ , if and only if the following two conditions hold:  $\forall a \in A(P_1) : pay-off(a, P_1) \geq pay-off(a, P_2)$  and  $\exists a \in A(P_1) : pay-off(a, P_1) > pay-off(a, P_2)$ .

### 3.3 Acceptable proposals

To define acceptable proposals, we introduce the basic concepts of Dung's abstract argumentation [8]. Dominance is not strong enough to reject a proposal, as several proposals can be accepted at the same time. We introduce now an *attack* relation among proposals such that a proposal attacks another proposal when accepting the former implies that the latter is not acceptable. Moreover, we say that a proposal attacks itself if the payoff for at least one of the agents is negative. Dung's theory offers the choice among several alternatives to define when an individual argument is acceptable. First we have to choose a semantics, then we have to choose whether the argument must be in the union or intersection of the extensions of this semantics. We say that an argument is acceptable if it is in the union of all admissible sets (which is the same as being in the union of the complete extensions, or the union of the preferred extensions).

**Definition 4 (Acceptable proposal)** A proposal argumentation framework (AF) is a pair  $\langle \mathcal{P}, \hookrightarrow \rangle$  where  $\mathcal{P}$  is a set of proposals called arguments and  $\hookrightarrow \subseteq \mathcal{P} \times \mathcal{P}$  is a binary attack relation over proposals, where  $P_1 \hookrightarrow P_2$  if and only if  $P_1$  dominates  $P_1 \cup P_2$ , or  $P_1 = P_2$  and the payoff of at least one of the agents is negative. Let  $C \subseteq \mathcal{P}$ . A set  $C$  is conflict-free if and only if there exist no  $P_i, P_j \in C$  such that  $P_i \hookrightarrow P_j$ . A set  $C$  defends an argument  $P_i$  if and only if for each argument  $P_j \in \mathcal{P}$  if  $P_j$  attacks  $P_i$  then there exists  $P_k \in C$  such that  $P_k$  attacks  $P_j$ . Let  $C$  be a conflict-free set of arguments, and let  $\mathcal{D} : 2^{\mathcal{P}} \mapsto 2^{\mathcal{P}}$  be the function such that  $\mathcal{D}(C) = \{P \mid C \text{ defends } P\}$ .  $C$  is admissible if and only if  $C \subseteq \mathcal{D}(C)$ . A proposal is acceptable, if and only if it is in some admissible set.

The example illustrates the central concepts of argumentation, reinstatement and dialogue games.

**Example 4 (continued)** The three proposals / ovals in Figure 1.4 may be called *ArthurBetty*, *ArthurCathy*, and *DanaCathy*. We have that *ArthurBetty* attacks *ArthurCathy*, and that *ArthurCathy* and *DanaCathy* attack each other. For example, *ArthurBetty* and *ArthurCathy* are conflicting, because Arthur needs only a single ride. *ArthurBetty* attacks *ArthurCathy* but not vice versa because *ArthurBetty* dominates *ArthurCathy*: Arthur prefers to ride with Betty. Moreover, he is the only agent involved in both proposals, so he has the power to choose which one he will investigate first.

Both *ArthurBetty* and *DanaCathy* are acceptable proposals. As Arthur will propose an agreement with Betty, Dana and Cathy can form another agreement. Using argument games, it is modelled as:

- Dana: Hi Cathy, I need to go to work. Can I get a ride with you?  
(Proposal for agreement *DanaCathy*.)
- Cathy: Hi, I am sorry Dana, but I planned to give Arthur a ride today ...  
(Proposal for alternative agreement *ArthurCathy*.)

- Arthur: Hmm...Actually, I really prefer riding with Betty.  
(*Proposal for alternative agreement ArthurBetty.*)
- *ArthurBetty is optimal so it is accepted*
- *ArthurCathy is therefore rejected*
- *DanaCathy no longer has an alternative, and is therefore accepted due to the reinstatement principle.*

The following proposition provides argumentation foundations for the do-ut-des principle. A proposal is exchange-based (or transaction-based) if every dependence is part of a *cycle*. This represents the fact that cooperation is based on reciprocity, called also *do-ut-des* [11]. If a reason occurs more than once in an abstract  $DN$ , it represents an OR-dependence. If it occurs more than once in a proposal, it is redundant.

**Definition 5 (Non-redundant proposals)** A cycle is a sequence of dependencies  $(s_1, t_1, d_1), \dots, (s_n, t_n, d_n)$  such that  $t_i = s_{i+1}$  for  $1 \leq i \leq n-1$ , and  $t_n = s_1$ . A proposal  $P \subseteq D$  is exchange-based if and only if there is a set of cycles  $C$  such that  $P = \cup C$ . Proposal  $P \subseteq D$  is non-redundant if and only if each reason occurs at most once.

**Proposition 1 (Do-ut-des)** If a proposal is acceptable, then it is exchange-based and non-redundant. If a proposal is exchange-based and non-redundant, then there are cost and benefit functions such that the proposal is acceptable.

**Example 5 (continued)** Proposals in Figure 1.4 are do-ut-des. A proposal that is not do-ut-des is the dependence relation  $\{(a : r_1, c : r_3, e_1)\}$ : Arthur shares Cathy's car, but does not give anything in return.

### 3.4 Minimal proposals

In this section, we define a suitable notion of minimal proposal. For example, if proposals consist of sub-proposals of disconnected components, then the agents can negotiate the sub-proposals one at a time.

**Definition 6 (Minimal proposal)** A proposal  $P$  is minimal iff it cannot be partitioned into two or more disjoint proposals  $P = P_1 \cup \dots \cup P_n$ , such that  $P$  is acceptable iff for  $1 \leq i \leq n$ :  $P_i$  is acceptable.

The following proposition provides argumentation foundations for Sauro [11]'s indecomposable do-ut-des property, abbreviated to i-dud. It shows that if a proposal is minimal, then it cannot be split into two sub-proposals sharing at most one agent.

**Proposition 2 (Indecomposable do-ut-des)** If proposal  $P$  is minimal, then there are no disjoint nonempty proposals  $P_1$  and  $P_2$  such that  $P = P_1 \cup P_2$  and  $P_1$  and  $P_2$  share at most one agent. If  $P$  is a proposal such that there are no disjoint nonempty proposals  $P_1$  and  $P_2$  such that  $P = P_1 \cup P_2$  and  $P_1$  and  $P_2$  share at most one agent, then there are cost and benefit functions such that  $P$  is minimal.

**Example 6 (continued)** Proposals in Figure 1.4 are i-dud. A proposal that is do-ut-des but not i-dud is the dependence relation  $\{(a : r_1, b : r_2, e_1), (b : r_2, a : r_1, e_4), (c : r_3, d : r_4, e_2), (d : r_4, c : r_3, e_3)\}$ .

## 4 Agreements on norms and roles

Norms and roles are mechanisms to obtain desired social behaviour, that is, to obtain desired agreements. We can use agreement technologies to obtain agreements on which norms are in force in the system, or which roles the agents play. In theory we could even consider higher level agreements, such as which meta-norms are in force to agree on which norms are in force. In the full model, we can use the agreement technologies to agree on the semantics for interoperability, or the trustworthiness of agents. In Figure 1, such agreements explain the role dependencies in Figure 1.2., and the role assignments in Figure 1.3.

Due to space limitations, we can only sketch the theory of higher levels agreements. Creating norms means that the set of acceptable agreements changes. The number of possible agreements can decrease, for example because sanctions make some agreements unacceptable, or increase, for example when an obligation becomes an incentive to start cooperating. Likewise, the assignment of roles to agents may increase or decrease the number of acceptable agreements.

## 4.1 Norm agreements

A social objective can be either a dependence relation that must be realized, called a positive objective and written as  $O^+$ , or a dependence relation that should not be realized, called a negative objective and written as  $O^-$ . For example, the agents are desired to help each other, not to crash into each other, etc.

**Definition 7 (System objective)** *Let  $DN = \langle N, E, D, c, b \rangle$  be a dependence network. The system objective  $O = \langle O^+, O^- \rangle$  with  $O^+, O^- \subseteq N \times N \times E$  is a pair of dependence relations.*

Norms are mechanisms used to obtain the social objective. We distinguish two kinds of norms. First, sanctions can be put on existing dependencies, such as penalties or taxes, and thus change only the cost and benefit functions. Second, obligations create new dependence relations, as the obligation of Betty in the running example. As with dependence networks and proposals for agreements, to model the norm creation problem, we start with a graph of all possible norms, and the actual role dependence is a sub graph.

**Definition 8 (Norm design)** *A norm network  $NN$  is a tuple  $\langle R, E, D^+, c^-, c^+, b^-, b^+ \rangle$  where  $R$  and  $E$  are two disjoint sets (of roles and reasons, respectively), and  $D^+ \subseteq A \times A \times R$  is a binary relation over roles for each reason such that  $(d_1, p_1, r)$  and  $(d_2, p_2, r)$  implies  $d_1 = d_2$ , and  $c^-, c^+, b^-, b^+ : A \times R \rightarrow \mathbb{R}^0$  are upper and lower bounds for the cost and benefit functions from roles and reasons to nonnegative real numbers. A role dependence network  $\langle R, E, D, c, b \rangle$  is a norm for  $NN$  if and only if  $D \subseteq D^+$ ,  $c^- \leq c \leq c^+$  and  $d^- \leq d \leq d^+$ .*

There are various ways to evaluate role networks against the system objective, or to compare role dependence networks with each other.

**Definition 9** *A role dependence network is acceptable if there is an acceptable proposal of the role playing agent dependence network containing  $O^+$  and not containing an element of  $O^-$ .*

## 4.2 Role assignment agreements

To model the role assignment problem, we can give as input a role network — set of possible role assignments — as a bipartite graph from agents or roles. The role network represents capabilities: who has the capabilities (diplomas, rank, etc.) to play a certain role.

**Definition 10 (Role network)** *A role network is a tuple  $RN = \langle A, R, PLAY \rangle$  where  $A$  and  $R$  are two disjoint sets (of agents and roles, respectively), and  $PLAY \subseteq A \times R$  is a binary relation. A relation  $play \subseteq A \times R$  is a role assignment for  $RN$  iff  $play \subseteq PLAY$  and each agent plays at most one role.*

The role assignment problem is analogous to the norm creation problem. Moreover, dialogues on norm creation and role assignment are analogous to dialogue in Example 4. How the three agreement processes interact is left for further research.

## 5 Summary and outlook

We introduce a uniform theory for arguing about agreements among role playing agents. Dependencies among role playing agents either derive from the dependencies among the agents, or from the dependencies among the roles the agents play. The reasons for dependences among agents are goals, tasks and resources,

and the reasons for dependences among roles are ought-to-be and ought-to-do obligations. We identify three kinds of agreements in our case study: agreements about car sharing, agreements about the norms in force, and agreements about role assignment. Due to space limitations, we only detailed the theory about the first kinds of agreements, and leave the details of the other agreements for future research. The theory about agreement establishes two kinds of results, one regarding admissibility, and another one regarding minimality. They make argumentation more efficient by focussing on the relevant alternatives. On the one side, *admissibility*: acceptable proposals are based on the do-ut-des principle: every agent gains something from the agreement. Likewise, acceptable role assignments and acceptable norms must have an effect on behavior, in the sense that acceptable role assignments and norms must create new cycles. On the other side, *minimality*: minimal proposals are based on the indecomposable do-ut-des principle: the proposal cannot be split into sub proposals. Likewise, minimal role assignments and norms should reach their goal effectively. Besides formalising the norm creation and role assignment agreements, in future research we will also extend our model to the other two layers of the agreement technologies tower.

## References

- [1] M. Abraham, D. M. Gabbay, and U. J. Schild. Obligations and prohibitions in talmudic deontic logic. *Artif. Intell. Law*, 19(2-3):117–148, 2011.
- [2] H. Billhardt, R. Centeno, C. E. Cuesta, A. Fernández, R. Hermoso, R. Ortiz, S. Ossowski, J. S. Pérez-Sotelo, and M. Vasirani. Organisational structures in next-generation distributed systems: Towards a technology of agreement. *Multiagent and Grid Systems*, 7(2-3):109–125, 2011.
- [3] G. Boella and L. van der Torre. Reasoning for agreement technologies. In *Proc. of ECAI*, pages 895–896, 2012.
- [4] P. Bresciani, A. Perini, P. Giorgini, F. Giunchiglia, and J. Mylopoulos. Tropos: An agent-oriented software development methodology. *JAAMAS*, 8:203–236, 2004.
- [5] P. Caire, L. van der Torre, and S. Villata. Argumentation theoretic foundations for abstract dependence networks. In *Proc. of AT*, 2013.
- [6] P. Caire, S. Villata, G. Boella, and L. van der Torre. Conviviality masks in multiagent systems. In *Proc. of AAMAS*, pages 1265–1268. IFAAMAS, 2008.
- [7] C. Castelfranchi, A. Cesta, R. Conte, and M. Miceli. Foundations for Interaction: The Dependency Theory. In *Proc. of AI\*IA, LNCS 728*, pages 59–64. Springer, 1993.
- [8] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.*, 77(2):321–358, 1995.
- [9] D. M. Gabbay. Fibring argumentation frames. *Studia Logica*, 93(2-3):231–295, 2009.
- [10] S. Ossowski, editor. *Agreement Technologies*. Law, Governance and Technology Series, Springer, 2013.
- [11] L. Sauro. Qualitative criteria of admissibility for enforced agreements. *CMOT*, 12(2-3):147–168, 2006.
- [12] J. S. Sichman and R. Conte. Multi-agent dependence by dependence graphs. In *Proc. of AAMAS*, pages 483–490. IFAAMAS, 2002.
- [13] J. van Benthem. *Logical Dynamics of Information and Interaction*. Cambridge University Press, 2011.
- [14] J. von Neumann and O. Morgenstern. *Theory of Games and Economic Behaviour*. Princeton Univeristy Press, 1944.
- [15] E. Yu. *Modelling Strategic Relationships for Process Reengineering*. PhD thesis, University of Toronto, 1995.