

Electronic institutions for B2B: dynamic normative environments

Henrique Lopes Cardoso · Eugénio Oliveira

Received: 17 October 2005 / Accepted: 13 March 2007 / Published online: 11 April 2007
© Springer Science+Business Media B.V. 2007

Abstract The regulation of the activity of multiple autonomous entities represented in a multi-agent system, in environments with no central design (and thus with no cooperative assumption), is gaining much attention in the research community. Approaches to this concern include the use of norms in so-called normative multi-agent systems and the development of electronic institution frameworks. In this paper we describe our approach towards the development of an *electronic institution* providing an enforceable normative environment. Within this environment, institutional services are provided that assist agents in forming cooperative structures whose commitments are made explicit through contracts. Our normative framework borrows some concepts from contract law theory. Contracts are formalized using norms which are used by the institution while monitoring agents' activities, thus making our normative environment dynamic. We regard the electronic institution as a means to facilitate both the creation and the enforcement of contracts between agents. A model of "institutional reality" is presented that allows for monitoring the fulfillment of norms. The paper also distinguishes our approach from other developments of the electronic institution concept. We address the application of our proposal in the B2B field, namely regarding the formation of Virtual Organizations.

Keywords Electronic institution · Multi-agent system · Norm · Contract

H. Lopes Cardoso (✉) · E. Oliveira
LIACC – NIAD&R, Faculty of Engineering, University of Porto,
R. Dr. Roberto Frias, Porto 4200-465, Portugal
e-mail: hlc@fe.up.pt

1 Introduction

The handling of business relationships between different entities, and particularly in the case of B2B settings, typically requires an explicit statement of each entity's responsibilities. This need is exacerbated when considering open settings, in which potential partners may not have a means to assess each others' previous performance reputations. Furthermore, in the real-world it is common to rely on institutions that establish some kind of social order and that enable the creation of commitments between different agents by governing a trustable environment. Similar concerns are being addressed within the multi-agent systems research community. In particular, there is an emphasis towards developing mechanisms that regulate multi-agent systems in environments with no central design (and therefore with no cooperative assumption). Thus, while agent theory describes agents as autonomous self-interested entities, preferably interacting in open environments, the application of MAS in real-world scenarios raises an important question: how to ensure cooperative behavior in scenarios populated with heterogeneous and self-interested agents?

At least two approaches to this problem may be identified (Castelfranchi 2000). One is to impose constraining infrastructures to agents, making it impossible for them to deviate from the desired behavior: this approach severely limits agent autonomy. The other possibility is to regulate the environment, providing incentives for cooperative behavior through normative constraints, and allowing agents to choose whether to obey or to violate them.

Normative multi-agent systems (Boella and van der Torre 2005) consider the latter case, in which agents are seen as entities that are subject to certain norms. After initial research on norms as constraints on behavior, it is now accepted that autonomous agents are able to deliberate whether to comply with norms (Castelfranchi et al. 2000); the environment influences their reasoning abilities by introducing incentives to cooperation (or discouraging deviation).

The concept of electronic institutions is being pursued, by some researchers, as the virtual counterpart of real world institutions. An electronic institution provides an environment that regulates the relationships between software agents. Some approaches have considered such an environment as a constraining infrastructure (Esteva et al. 2002), where the institution imposes the actions that agents may perform, thereby defining an interaction protocol that agents must follow. We do not follow such a restrictive scenario.

In our approach, the *Electronic Institution* (EI) concept has two main purposes:

- (1) to support agent interaction as a coordination framework, making the establishment of business agreements more efficient; and
- (2) to provide a level of trust by offering an enforceable normative environment.

Therefore, our perspective regards an EI not as an end per se, but as a means to facilitate both the creation and the enforcement of contracts between agents. This is accomplished by providing a number of agent-based institutional services (Lopes Cardoso et al. 2005), such as negotiation mediation (Oliveira and

Rocha 2000), ontology mapping (Malucelli et al. 2005), and contract monitoring and enforcement (Lopes Cardoso and Oliveira 2005a).

Furthermore, we consider the EI as an environment within which agents can form virtual organizations (VO). These consist of consortiums defining cooperation efforts between agents, and involve specific interactions during a certain time frame. VO agreements and their operation are formalized through contracts. In a VO setting, agents may represent different business units or enterprises, which come together to address new market opportunities by combining skills, resources, risks and finances no partner can alone fulfill (Dignum and Dignum 2002). The subject of virtual organizations is gaining increasing importance in the B2B world, where players are becoming more focused on their core businesses and rely on outsourcing and dynamic consortiums. This can lead to complex relationships, in which partners' compliance must be assessed.

The EI encompasses a set of norms regulating the environment. However, as agents negotiate towards the achievement of agreements formalized in contractual norms, this normative environment is dynamic. Through appropriate services, the EI monitors and enforces both institutional norms and those formalizing contracts.

The EI is thus a regulated environment where agents interact. In order to enforce norms, the EI must have a means of acknowledging what is going on. When considering software agents, interactions are based on the exchange of messages (illocutions). Inside the EI, agents' illocutions are the starting point towards the formation of *institutional reality* (inspired by Searle (1995)), together with institutionally certified roles that agents may perform. This reality embraces the occurrence of actions regarding the compliance to institutional and contractual norms; it also takes into account elements related to the task of processing norms, mapping its semantics.

Within institutional boundaries, some agents perform specific *institutional roles*, being certified by the EI as legitimate to achieve certain institutional facts. External agents may also announce themselves as performing certain external roles. Instead of providing institutional services, these are general roles (such as supplier or customer) which may, when performed inside the EI, have a set of attached norms. By assuming those roles, agents become committed to these norms.

In this paper we develop on our approach towards the development of an Electronic Institution that provides an environment where norms are monitored and enforced. In Sect. 2 we describe our framework of norms, classifying them according to their scope. Section 3 addresses the issue of managing "institutional reality," by detailing how it is created and how to define rules that manipulate it. Section 4 introduces our norm representation formalism and the definition of rules for norm monitoring; we also include examples of contract representations. Section 5 describes implementation issues of our system. Section 6 concludes and relates our work with other authors.

2 An institutional framework of norms

The social sciences identify the concept of social structure as an underlying framework of norms that imposes a sense of order and predictability within a group of individuals. These norms are typically attached to roles played by members of a society. A role represents the way someone is expected to behave in particular situations, by having an associated set of normative expectations.

Different types of norms can be identified, according to their abstraction level. While at one hand we have values, conventions, and abstract norms that are implicitly adhered to and may not have a formally defined social response in case of deviation, on the other hand we have formal and legal norms that include explicitly defined punishments to deal with infractions.

Within a society, different entities may also choose to build social relationships (e.g., in the business field) that are to be regulated by appropriate norms. Contracts are signed that explicitly state what behavior is expected from each involved party. Specific institutions exist (e.g., the court of law) that allow for enforcing those contracts.

Norms also play an important role in open artificial agent systems; they have been said to improve coordination and cooperation (Conte et al. 1999). As in real-world societies, norms provide us a way to achieve social order (Castelfranchi 2000) by controlling the environment and making it more stable and predictable. Our overall institutional picture considers the EI as an environment imposing a set of institutional norms, but also allowing agents to create mutual commitments by voluntarily adhering to a set of norms that make those commitments explicit. The EI will thus act as a trustable third-party that receives contracts to be monitored and enforced. Therefore, instead of acting as recourse entity to which agents can turn to in case of conflicts, the EI proactively assists contract execution and automatically employs necessary measures when norms are disrupted.

Considering the stated scenario, inside the EI we may have different norms applicable in different contexts, and which therefore have different scopes. Furthermore, with the intent of facilitating contract formation, we approach the normative framework using a layered and hierarchical approach, enabling the adoption of contract law concepts such as the notion of “default rules” (Craswell 2000).

Our norm classification considers three normative layers (Lopes Cardoso and Oliveira 2005b): institutional, constitutional, and operational.

Institutional norms regulate the behavior of every agent inside the EI. As behavior prescription is typically dependent on agents’ commitments, institutional norms represent the commitments of agents towards the EI as a whole: by assuming roles (that is, by signing a “social contract”), agents become committed to their associated norms. Institutional norms set up the normative background on which cooperation commitments can be established.

Constitutional norms are used to describe the foundation of agents’ virtual organizations, which thereby commit to a certain cooperation agreement. The terms of such an agreement are specified by means of norms regulating the created

consortium, which usually exists for a period of time. These norms have a narrower scope, since they apply only to the agents participating in the VO.

Finally, *operational norms* specify contracts by indicating actions to be performed by contractual agents. Operational contracts may be proposed and signed within the context of a specific VO, involving a subset of the VO partners; they may also comprise a stand-alone deal (in which case they can simply be referred to as contractual norms). These contracts have a single-business nature, whereas VO contracts may comprise a long-term alliance.

Figure 1 depicts the organization of norms inside the EI, as we have just described. The links between the different shapes indicate hierarchical relationships that allow for norm inheritance, as described below.

The above description may suggest that different types of norms are created at different moments. Thus, institutional norms may be pre-existent, while constitutional norms are created when agents reach a cooperation agreement, and operational norms come into existence when executable contracts are signed. However, norms with limited scopes may be predefined for a number of reasons, as follows.

An important concept in contract law theory is the use of “default rules” (Craswell 2000). These exist with the intent of facilitating the formation of contracts, allowing them to be underspecified by defining default clauses or default values. Therefore, constitutional or even operational norms may be institutionally defined: together with institutional norms, they provide a normative background in which agents can rely to build their contractual commitments. Furthermore, just as real-world legislations are organized through hierarchies of laws, it is natural to have predefined regulations devoted to particular contexts, such as the VO setting. Norms regulating specific aspects of this type of contracts are naturally predefined. Agents can rely on these regulations as a ground basis to raise specific virtual organizations.

According to this setting, it is possible, therefore, to predefined scoped norms that are to be imposed when the activity they regulate is adhered to by agents. Although having a limited scope, these norms can be seen as institutional in the sense that they are institutionally predefined.

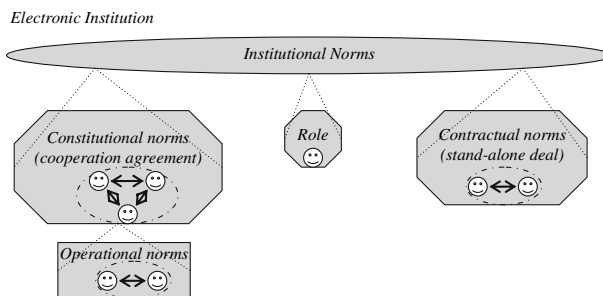


Fig. 1 Norms in an electronic institution

In any case, norms define, within a given context, what ought to be done in certain circumstances. While we defer a proper formalization of norm representation to Sect. 4, this can be briefly stated as follows:

$$[\textit{Context}] \textit{Situation} \rightarrow \textit{Prescription}$$

The *Context* indicates the scope of the norm: as such, we may have norms applicable inside the whole institutional environment, while other norms may be defined, e.g., inside a particular agents' VO. The *Situation* describes when the norm is in place. The *Prescription* specifies what should be accomplished in order for the norm to be fulfilled.

Using this representation, norms obviously lend themselves to a rule-based implementation. The situation part refers to the rule conditions, whereas the prescription part corresponds to the rule's effects. Norms can be seen as contextual rules that prescribe behavior.

3 Managing institutional reality

We envision the EI as an open environment, consisting of a place where social relationships are created and enforced. The interaction between autonomous agents is supervised by the institution, which detects the appearance of social interdependencies (commitments) between agents, which must be enforced. The operation of the EI embraces the notion of *institutional reality* (inspired by Searle (1995)). This reality is in part constructed by attributing institutional semantics to agent interactions.

Furthermore, as agents' commitments are expressed through norms, it is essential to provide means that allow the EI to monitor norm compliance. In particular, it is important to grasp the obligations that are in place, and to take into account both their fulfillment and violation, considering the passage of time. These elements of institutional reality are important for norm practicability.

3.1 Brute facts and institutional facts

Following Searle's theory on "the construction of social reality" (Searle 1995), we distinguish between *brute facts* and *institutional facts*. The latter are obtained from the former, through rules defining "counts-as" relations (*constitutive rules*, according to Searle 1995; *empowerments*, according to Jones and Sergot 1996). In our case, brute facts refer to agents' actions. In a society of artificial software agents, the only observable actions are the utterance of illocutions: these are our basic building blocks, that is, the brute facts we build on to infer institutional reality. Rules regulating the creation of institutional facts will verify not only what is uttered, but also who perpetrates such actions—the effects of illocutions may depend on who utters them. While Searle considers that constitutive rules may iterate through institutional facts to infer new ones, we prefer to consider this possibility separately through institutional rules, as explained in Sect. 3.3.

The following EBNF definitions introduce the concepts of *role*, *brute fact* and *institutional fact*.

<AgentRole>	::=	agentRole(<Agent>, <Role>)
<Agent>	::=	agent_identifier
<Role>	::=	role_identifier
<BruteFact>	::=	illocution(<Sender>, <Content>)
<Sender>	::=	<Agent>
<Content>	::=	atomic_formula_based_on_institutional_ontology
<InstitutionalFact>	::=	[“[’<Ctx>’]”]ifact(<IFact>, <Timestamp>)
<Ctx>	::=	context_identifier
<IFact>	::=	atomic_formula_based_on_institutional_ontology
<Timestamp>	::=	temporal_reference

Role definitions allow us to declare which agents perform which institutional roles. These roles will have constitutive rules attached that indicate their powers in terms of achieving institutional facts.

Brute facts represent agents’ illocutions. Considering that agents use the FIPA-ACL message structure specification (FIPA 2002a), the sender and content components refer to the homonym ACL message parameters. Illocutions uttered by agents are assumed to use a well-defined institutional ontology, which enables their use in constitutive rules. Note that these illocutions are messages sent to the institution as the “normative authority,” with the intent of informing it that something happened concerning a certain relationship supervised by the institution.

Institutional facts are institutional reality’s elements. Each institutional fact refers to a certain context. The main context consists of the EI itself, inside which certain illocutions “count as” certain institutional facts. However, if we consider facts denoting agent behavior related to the norms it must abide to (either originated by the agent’s role or by a certain contract), these facts occur within the context of the norm they report to (more on this in Sect. 4.2). An important issue to consider in contracting scenarios is time. Every fact occurs at a given instant, corresponding to a synchronized institutional clock.

The transformation of brute facts into institutional facts is done by applying *constitutive rules*. Institutionally relevant facts include those events that are important in respect to commitment fulfillment. Therefore, rules regulating how these facts come about are needed.

3.2 Constitutive rules

Constitutive rules make a connection between what is said and what is taken for granted. Constitutive rules take into account a set of *institutional roles* enacted by agents providing certain services. Therefore, some institutional facts may come into existence only if agents performing certain institutional roles execute some predetermined actions (that is, utter appropriate illocutions). Authoritative relations are thus established between roles and institutional reality: an agent performing a

given role is said to be empowered (as in Jones and Sergot 1996) to achieve the effects expressed in its role-related constitutive rules.

The following EBNF definitions introduce the concept of constitutive rule:

$$\begin{aligned} \langle \text{ConstitutiveRule} \rangle & ::= \langle \text{BruteFacts} \rangle \{ \text{“}\wedge\text{”} \langle \text{AgentRole} \rangle \} \\ & \quad \text{“}\rightarrow\text{”} \langle \text{InstitutionalFacts} \rangle \\ \langle \text{BruteFacts} \rangle & ::= \{ \langle \text{BruteFact} \rangle \text{“}\wedge\text{”} \} \langle \text{BruteFact} \rangle \\ \langle \text{InstitutionalFacts} \rangle & ::= \{ \langle \text{InstitutionalFact} \rangle \text{“}\wedge\text{”} \} \langle \text{InstitutionalFact} \rangle \end{aligned}$$

The semantics of constitutive rules is straightforward: when specific pieces of “brute reality” come about, certain pieces of “institutional reality” can be inferred. Further iterations through institutional facts are captured with institutional rules, as explained in Sect. 3.3.

Constitutive rules must inspect brute facts in order to obtain the corresponding institutional facts. This corresponds to mapping between two different sub-ontologies: one for brute and another for institutional facts. Although the institutional facts’ ontology must be well defined in order for facts to be used in norms and rules, it is possible to devise a scenario with multiple sub-ontologies for brute facts. Constitutive rules implicitly define which terms are accepted and recognized, as they “filter” the illocutions that are uttered transforming them into significant institutional facts.

3.2.1 Examples of constitutive rules

Constitutive rules are important to allow the recognition of action execution. This includes the fulfillment of contractual obligations through the realization of certain transactions. As we are concerned with the application of our model to business scenarios involving transactions, we identify three main institutional roles providing a connection to the real-world: a *banking* role enables acknowledging monetary value transfers; a *delivery tracker* role certifies product delivery; a *messenger* role provides certified information exchange facilities.

In the following illustrations, variables are represented with identifiers starting with the question mark (?) character.

Consider a situation in which an agent ought to make a certain payment to another. Although the agent may claim to have paid its debt, that does not make it the case. Still, if an independent financial third party agent, providing a certified institutional service, states that a currency transfer referring to a certain context (e.g., a purchase contract) has taken place, it would be safe to consider that the payment took place:

$$\begin{aligned} & \text{illocution}(?B, \text{currency_transferred}(?Ctx, ?Ag1, ?Amount, ?Ag2, ?Time)) \wedge \\ & \text{agentRole}(?B, \text{bank}) \\ & \rightarrow [?Ctx] \text{ifact}(\text{payment}(?Ag1, ?Amount, ?Ag2), ?Time) \end{aligned}$$

We can also say that if both agents (the payer and the receiver) state that a payment took place, it would also be safe to conclude the associated institutional fact:

$$\begin{aligned} & illocution(?Ag1, paid(?Ctx, ?Amount, ?Ag2, ?)) \wedge \\ & illocution(?Ag2, collected(?Ctx, ?Amount, ?Ag1, ?Time)) \\ & \rightarrow [?Ctx]ifact(payment(?Ag1, ?Amount, ?Ag2), ?Time) \end{aligned}$$

If we aim at certifying product delivery, we may trust on a delivery tracking service:

$$\begin{aligned} & illocution(?DT, delivered(?Ctx, ?Ag1, ?Item, ?Quantity, ?Ag2, ?Time)) \wedge \\ & agentRole(?DT, delivery_tracker) \\ & \rightarrow [?Ctx]ifact(delivery(?Ag1, ?Item, ?Quantity, ?Ag2), ?Time) \end{aligned}$$

And again rely on the confirmation of both involved parties:

$$\begin{aligned} & illocution(?Ag1, sent(?Ctx, ?Item, ?Quantity, ?Ag2, ?)) \wedge \\ & illocution(?Ag2, received(?Ctx, ?Item, ?Quantity, ?Ag1, ?Time)) \\ & \rightarrow [?Ctx]ifact(delivery(?Ag1, ?Item, ?Quantity, ?Ag2), ?Time) \end{aligned}$$

The same methodology can be applied concerning the exchange of messages. If message delivery recognition is a must, a messenger role may provide such a service. This way, interactions between agents through the exchange of messages can be recorded, as long as they are intermediated. Following FIPA specifications, agents may use this service by using the FIPA-ACL *proxy* communicative act (FIPA 2002b). Another possibility is to consider this as the Message Transport Service (FIPA 2002c) of an agent platform. In any case, the messenger informs the EI that a given message was delivered. The following constitutive rule applies:

$$\begin{aligned} & illocution(?M, msg_delivered(?Ctx, ?Ag1, ?Msg, ?Ag2, ?Time)) \wedge \\ & agentRole(?M, messenger) \\ & \rightarrow [?Ctx]ifact(msg(?Ag1, ?Msg, ?Ag2), ?Time) \end{aligned}$$

By defining institutional roles instead of institutional agents providing their associated services, we emphasize the open and distributed nature of our institutional environment. Therefore, we may have several agents performing the same institutional role, and thus providing the same institutional service. By “institutional” we mean that those agents are certified by the EI as being trustworthy.

3.3 Institutional rules

The government of the institutional environment must take into account the dynamic nature of institutional reality. As described above, institutional facts may be achieved from agent illocutions, which are processed through constitutive rules. However, other elements, not directly dependent on illocutions, compose the reality that the EI is supposed to monitor. Examples include pending obligations, their fulfillment and violation. Also, the passage of time is important to verify the compliance to norms. All these elements may have interdependencies that may be

made explicit by defining *institutional rules*. These rules, working only on institutional reality elements, are thus different from the constitutive rules defined above.

The EBNF description of institutional reality elements follows (besides institutional facts, which we have introduced previously):

```

<Obligation> ::= [“[“<Ctx>“]”]obligation(<Agent>, <IFact>, <Deadline>)
<Fulfillment> ::= [“[“<Ctx>“]”]fulfilled(<Agent>, <IFact>, <Timestamp>)
<Violation> ::= [“[“<Ctx>“]”]violated(<Agent>, <IFact>, <Timestamp>)
  <Time> ::= [“[“<Ctx>“]”]time(<Timestamp>)
  <Deadline> ::= <Timestamp>[“+”time_units]

```

As with institutional facts, obligations, fulfillments and violations may refer to a particular context. Obligations indicate that a certain agent (the obligation’s bearer) is expected to bring about a certain institutional fact before a certain deadline. Fulfillment elements register the observance of obligations, while violation elements signal their infringement. Both of these elements include a timestamp indicating when the obligation was considered as fulfilled/violated. Time instances make time passage discrete by signaling instants that are relevant to the context where they are defined. It is the institution’s responsibility to generate such ticks.

Institutional rules work on these elements in order to explicitly define their interdependencies:

```

<InstitutionalRule> ::= [“[“<Ctx>“]”]<Antecedent>“→”<Consequent>
  <Antecedent> ::= {<Prem>“^”}<Prem>{“^”¬<Prem>}
  <Consequent> ::= {<Conc>“^”}<Conc>
    <Prem> ::= <InstitutionalFact>|<Obligation>|<Fulfillment>|
              <Violation>|<Time>|<RelCondition>
    <Conc> ::= <InstitutionalFact>|<Fulfillment>|<Violation>|
              institutional_procedure
  <RelCondition> ::= <Operand><RelOperator><Operand>
    <Operand> ::= variable|expression
    <RelOperator> ::= “<”|“>”|“≤”|“≥”|“=”

```

The semantics of these rules is straightforward: whenever their antecedent part verifies, the consequent part holds. The point of associating a context with a rule is that the context works as a wrapper for all formulae within that rule. Besides institutional reality elements, the rule’s conditions may include relational expressions, useful e.g., for comparing time tags. As for the rules’ consequents, they can consist of any institutional reality elements but obligations. As we explain in the following section, this is what distinguishes rules from norms: the latter allow us to prescribe behavior by imposing certain obligations. We also allow institutional rules

to point to procedures not amenable to a declarative representation. For instance, we may define a rule that triggers a notification procedure whenever a new obligation arises.

It is worth pointing out that institutional rules allow us to obtain new institutional facts from existing ones. This iteration is part of Searle's model for constitutive rules; however, we find it useful to allow for such further inferences to be contextualized. That is, while the creation of institutional reality from outside elements (brute facts) is made through constitutive rules whose scope is always the institution itself, the "counts as" formula for institutional facts may be context-dependent—the institutional rule will define the scope. This way, it may be the case that only in certain contractual relationships an institutional fact (as obtained from appropriate brute facts) is enough to conclude another one, and this association may be defined e.g., by matters of trust or business specificities. By allowing these rules to be context specific, we therefore assure the expansibility of the system.

In the next section we exemplify the use of institutional rules in the task of monitoring the compliance to norms.

4 Norm specification and enforcement

Having as one of its main goals the establishment of an enforceable normative environment, it is necessary to define, inside the EI, how to represent norms and furthermore how to monitor their compliance. In other words, besides describing how norms are to be represented, it is necessary to define what conditions need to be verified in order to conclude that a norm has been violated or fulfilled.

A norm-aware environment can operate either preventively, by making unwanted behavior impossible, or reactively, by detecting violations and reacting accordingly (Vázquez-Salceda et al. 2004). Taking into account the autonomous nature of agents, we rely essentially on the latter practice.

Norms prescribe the expected behavior of agents, specifying states of affairs that *must* be brought about by an agent before a certain deadline. Norms allow us to affect the normative positions of the agents in the system (Fornara et al. 2005). We consider *obligations* as the means to express the prescription of behavior norms. The pending obligations are expressed as elements of institutional reality, as explained in the previous section. Instead of dictating the exact action an agent must perform, obligations prescribe the institutional fact that he must bring about. This fits with our model of institutional reality, where we specify by means of constitutive and institutional rules how institutional facts may be accrued. It also enables an agent to delegate or outsource tasks conducting to the accomplishment of such state of affairs, while still being responsible before the institution for the (un)fulfillment of the obligation.

The definition for norms follows:

$$\begin{aligned}
 \langle \text{Norm} \rangle & ::= [“[” \langle \text{Ctx} \rangle “]”] \langle \text{Situation} \rangle “\rightarrow” \langle \text{Prescription} \rangle \\
 \langle \text{Situation} \rangle & ::= \{ \langle \text{Cond} \rangle “\wedge” \} \langle \text{Cond} \rangle \{ “\wedge” \neg \langle \text{Cond} \rangle \} \\
 \langle \text{Prescription} \rangle & ::= \{ \langle \text{Obligation} \rangle “\wedge” \} \langle \text{Obligation} \rangle \\
 \langle \text{Cond} \rangle & ::= \langle \text{InstitutionalFact} \rangle | \langle \text{Fulfillment} \rangle | \langle \text{Violation} \rangle | \langle \text{Time} \rangle | \\
 & \quad \langle \text{RelCondition} \rangle | \langle \text{ContextualInfo} \rangle \\
 \langle \text{ContextualInfo} \rangle & ::= “[” \langle \text{Ctx} \rangle “]” \textit{atomic formula dependent on context type}
 \end{aligned}$$

Norms prescribe behavior by specifying what obligations come about when a specific situation is accomplished. The situation is characterized by any institutional reality element, and can also access information that is context-dependent. As with institutional rules, the point of associating a context with a norm is that it relieves us from indicating that context for every norm component (describing the situation or the prescription part).

4.1 Detecting the fulfillment and violation of obligations

The monitoring of norm compliance is done by defining appropriate institutional rules. These will make a connection between institutional facts and the obligations they may relate to. In the following examples, variables are represented with identifiers starting with the question mark (‘?’) character.

The fulfillment of obligations is verified using a *fulfillment rule* applicable to all contexts (that is, to any contractual relationship monitored by the ED):

$$\begin{aligned}
 [?Ctx] \textit{ifact}(?IFact, ?T) \wedge \textit{obligation}(?Agent, ?IFact, ?Deadline) \wedge ?T < ?Deadline \\
 \rightarrow \textit{fulfilled}(?Agent, ?IFact, ?T)
 \end{aligned}$$

This rule indicates that if an institutional fact prescribed by an obligation is achieved before its deadline, then that obligation is fulfilled. All rule’s literals are contextualized. That is, if an obligation within a certain contract is accomplished, the fulfillment of such obligation occurs, obviously, inside that same contract. However, this rule applies to all contractual relations (it is an institutional rule); it thus has un-instantiated context.

This rule is fundamental for enabling the chaining of obligations within a contractual relationship. It establishes a connection between the institutional facts that are added and the pending obligations.

Violation detection is achieved by an institutional rule that fires when deadlines have elapsed. For this we consider time events, which are generated by the electronic institution corresponding to the time when obligations are due:

$$\begin{aligned}
 [?Ctx] \textit{time}(?Deadline) \wedge \textit{obligation}(?Agent, ?IFact, ?Deadline) \wedge \\
 \neg \textit{fulfilled}(?Agent, ?IFact, ?) \\
 \rightarrow \textit{violated}(?Agent, ?IFact, ?Deadline)
 \end{aligned}$$

This violation detection rule states that in any context, if a deadline referring to an obligation was reached, and such obligation was not fulfilled, then a violation occurred (in the same context).

The resulting fact stating that a violation has occurred may be used to apply punishment measures, either direct (sanctions, restrictions, interdictions) or indirect (reputation). Thus, we may define sanctioning norms, that is, norms prescribing obligations in case of violations (also known as contrary-to-duty obligations (Jones and Carmo 2001)):

$$[\langle Ctx \rangle] \text{violated}(?Agent, \langle IFact \rangle, \langle Deadline \rangle) \\ \rightarrow \text{obligation}(?Agent, \langle IFact \rangle, \langle Deadline \rangle)$$

This type of norm states that if the agent identified as the bearer for the achievement of an institutional fact fails to do so by a certain deadline, then a new obligation (presumably harder) is assigned to him. Placeholders for specifying this general sanction norm template were left with angle brackets.

We may also define institutional rules that apply some institutional procedure concerning an agent's reputation:

$$[?Ctx] \text{violated}(?Agent, ?IFact, ?) \rightarrow \text{update_reputation}(?Agent, ?IFact)$$

In this example, we consider that any kind of violation should lead to a reputation update. The un-instantiated context also indicates the broad applicability of this rule.

Using a similar approach, one could also define prohibitions based on violations. Institutional procedures may be invoked that interdict agent access to institutional services, such as message delivery, negotiation mediation, or contract registration through a notary service.

4.2 Organizing contexts

Our hierarchical framework of norms demands for their organization through contexts. In order to achieve such an organization, we define each contractual structure as a context. Therefore, a VO is a context, and so is an operational or a social contract. Each of these contractual situations pertains to another context, from which it may inherit norms.

We find it useful to define a general means of representing a context:

$$\begin{aligned} \langle Context \rangle & ::= [“[” \langle Ctx \rangle “]”] \text{context}(\langle ContextType \rangle, \langle Ctx \rangle, \langle Who \rangle, \\ & \quad \langle Timestamp \rangle) \\ \langle ContextType \rangle & ::= \text{type.of.context} \\ \langle Who \rangle & ::= \langle Agent \rangle \{ “^” \langle Agent \rangle \} \end{aligned}$$

A context will include a context type and an identifier, and also information about who is involved and when it was created.

The context type indicates what norms can be inherited, that is, which “default rules” may be applicable. This allows us to define default clauses that are to be applied to contexts of a certain type. For instance, we may define that by default a certain clause should be applicable to all purchase contracts inside the EI. On the other hand, the context’s (super) context indicates from where norms can be inherited. For instance, we may have an operational contract that is contextualized in a certain VO, and therefore should be able to inherit norms from it. Should there be no applicable norms defined at this level, norms from the institution (the VO’s super-context) should be inherited. It is the normative environment’s responsibility to use this structured context representation in order to find applicable norms in each situation.

For ease of illustration, we make the assumption that the context identifier, represented above as <Ctx>, indicates both its type and its unique id, in the form <ContextType>:<Id>. Further contextual information (e.g., the participants roles, resources and prices to be exchanged, and so on) is explicitly stated as explained before.

4.3 Examples

In this subsection we give some examples of norm specification in particular contracting situations. Throughout the examples, placeholders for the specific details of norms or rules are left with angle brackets. A variable appearing within angle brackets indicates that its value is that of its previous appearance in another rule or norm.

4.3.1 Governing virtual organization cooperation agreements

A cooperation agreement aggregates the organization’s constitutional information, including the cooperation effort agents commit to. As an exemplifying scenario, we consider a situation in which the intended cooperation consists on the exchange of resources: each partner commits to the effort of supplying, upon request, a certain quantity of a given resource to another partner, with an associated and agreed price.

First, the context definition and the contextual information, which may be defined through the following templates:

```
context(coop_agreement, coop_agreement : <Id>, <Who>, <Timestamp>)
[coop_agreement : <Id>]coop_effort(<Agent>, <Resource>,
    <MaxQt>, <UnitPrice>)
[coop_agreement : <Id>]business_process(<Agent>, <Resource>, <Agent>)
```

The *coop_effort* information indicates that an agent is willing to respond to orders of at most <MaxQt> units with an associated price <UnitPrice>. The *business_process* element is useful when considering multiple partners, allowing us to define what resources should flow between which agents. Now, we may add a norm indicating that the agent is obliged, upon request, to stick to his promise:

$$\begin{aligned}
& [coop_agreement : ?IdCA] \text{ ifact}(msg(?A1, request(?Res, ?Qt), ?A2), ?TReq) \wedge \\
& coop_effort(?A2, ?Res, ?MaxQt, ?) \wedge business_process(?A2, ?Res, ?A1) \wedge \\
& \quad ?Qt \leq ?MaxQt \\
& \rightarrow obligation(?A2, msg(?A2, confirm(?Res, ?Qt), ?A1), ?TReq + 10)
\end{aligned}$$

This norm is here defined as applicable to all cooperation agreements of this kind (it is a “default rule”), hence its un-instantiated context. We then connect a VO with its operation, by defining an institutional rule indicating when new operational contracts are created:

$$\begin{aligned}
& [coop_agreement : ?IdCA] \text{ fulfilled}(?A2, msg(?A2, confirm(?Res, ?Qt), ?A1), \\
& \quad ?TConf) \wedge coop_effort(?A2, ?Res, ?, ?UnitPr) \\
& \rightarrow register_new_op_contract(?IdCA, ?A1, ?A2, ?Res, ?Qt, ?UnitPr, ?TConf)
\end{aligned}$$

This rule is also applicable to all cooperation agreements of this kind. The institutional procedure *register_new_op_contract* is responsible for creating the new operational context and its associated information:

$$\begin{aligned}
& [coop_agreement : ?IdCA] \\
& context(op_contract, op_contract : <?Id>, <?A1> \wedge <?A2>, <?TConf>) \\
& [op_contract : <?Id>] start(<?TConf>) \wedge purchaser(<?A1>) \wedge supplier(<?A2>) \wedge \\
& resource(<?Res>) \wedge quantity(<?Qt>) \wedge price(<?UnitPr>)
\end{aligned}$$

Norms regulating the operational contracts that VO partners may create need to be defined. If such norms are created within each operational contract, we may have:

$$\begin{aligned}
& [op_contract : <?Id>] time(<?TConf>) \\
& \rightarrow obligation(<?A2>, delivery(<?A2>, <?Res>, <?Qt>, <?A1>), \\
& \quad <?TConf> + 10) \\
& [op_contract : <?Id>] fulfilled(<?A2>, delivery(<?A2>, \\
& \quad <?Res>, <?Qt>, <?A1>), ?TDeliv) \\
& \rightarrow obligation(<?A1>, payment(<?A1>, <?UnitPr> * <?Qt>, <?A2>), \\
& \quad ?TDeliv + 30)
\end{aligned}$$

These norms state that a delivery obligation is in place with a deadline of 10 time units after contract creation; when this obligation is fulfilled, the purchaser is obliged to make the associated payment within 30 time units. The VO constitution could also define these norms as “default rules” for all operational contracts within this VO; for this, we just need to un-instantiate their contexts’ ids, and refer to the contextual information of the operational contract in order to get the variables’ values.

4.3.2 Governing purchase contracts

Purchase contracts represent a simpler case of contractual norms, in that they implement a single business operation. The following context templates may be defined:

$$\begin{aligned} & context(purchase_contract, purchase_contract : \langle Id \rangle, \langle Agent \rangle \\ & \quad \wedge \langle Agent \rangle, \langle Timestamp \rangle) \\ & [purchase_contract : \langle Id \rangle] start(\langle Timestamp \rangle) \wedge vendor(\langle Agent \rangle) \wedge \\ & customer(\langle Agent \rangle) \wedge item(\langle Item \rangle) \wedge price(\langle Price \rangle) \end{aligned}$$

The contract may then be executed complying with the following norms:

$$\begin{aligned} & [purchase_contract : ?Id] start(?S) \wedge time(?S) \wedge vendor(?V) \\ & \quad \wedge customer(?C) \wedge item(?I) \\ & \rightarrow obligation(?V, delivery(?V, ?I, 1, ?C), ?S + 2) \\ & [purchase_contract : ?Id] fulfilled(?V, delivery(?V, ?I, 1, ?C), ?TDeliv) \wedge price(?P) \\ & \rightarrow obligation(?C, payment(?C, ?P, ?V), ?TDeliv + 3) \end{aligned}$$

Again, these norms are defined as “default rules,” hence their un-instantiated context. Should a purchase contract have a different approach, norms must be defined that override these default regulations.

4.3.3 Governing social contracts

As introduced in Sect. 2, norms can be defined that regulate the activity of agents performing certain roles within the institution’s boundaries. The contracts established between those agents and the EI will be the context of such norms.

For instance, we may represent the seller role of an agent through the following context template:

$$\begin{aligned} & context(seller_social_contract, seller_social_contract : \langle Id \rangle, \\ & \quad \langle Agent \rangle, \langle Timestamp \rangle) \end{aligned}$$

And add the following contextual information:

$$[seller_social_contract : \langle Id \rangle] seller(\langle Agent \rangle)$$

We may then have norms that apply to all agents announcing themselves as sellers. Typically, these norms will be predefined as institutional and applicable to every seller role enactment. For instance, we may impose that every seller should answer to a “request-for-quotes” within a reasonable time period:

$$\begin{aligned} & [seller_social_contract : ?Id] ifact(msg(?Ag1, request_quote(?X), \\ & \quad ?Ag2), ?T) \wedge seller(?Ag2) \\ & \rightarrow obligation(?Ag2, msg(?Ag2, quote(?X, ?Price), ?Ag1), ?T + 2) \end{aligned}$$

This approach enables us to use a consistent representation of norms within the EI, considering both norms attached to roles and norms that apply to contractual agreements.

5 Implementing an institutional normative environment

Most work on normative systems relies on deontic formalizations, and focuses on defining formal semantics. However, recent attention is being given towards norm implementation in MAS (some guidelines can be found in Vázquez-Salceda et al. 2004). An explicit declarative representation of norms is needed. If norms are enclosed inside agents' code, we get a static environment where norms cannot change over time. This prevents agents from adopting norms by establishing commitments through contracts.

Our normative approach relies essentially on obligation prescription. Since obligations have associated deadlines, an institutional clock is essential for violation detection. It is responsible for generating institutional reality elements corresponding to time instants when deadlines are due.

Fulfillment detection is based on the assumption that it is in the best interest of agents to publicize the fulfillment of obligations. They do so by provoking the achievement of corresponding institutional facts (which are accrued by authorized agents). If agents fail to convince the EI that they complied, then they expose themselves to punishments, either direct (e.g., sanctions) or social (e.g., reputation records). This means that the active part of norm enforcement concerns the detection of violations, while their fulfillment is verified passively, upon the occurrence of external events (that is, events triggered by external agents).

Not surprisingly, both rules (either constitutive or institutional) and norms can be easily implemented with a rule-based approach. Since the normative environment is based on the occurrence of events, the obvious solution towards its implementation is by using a forward-chaining production system.

Therefore, we are pursuing the development of a first prototype using the Jess shell (Friedman-Hill 2003). Jess is a rule engine that very efficiently applies rules to data. Our knowledge base consists of rules and norms. The working memory includes the elements of institutional reality. When those elements are obtained as a result of applying a certain rule or norm, they are asserted in working memory. Figure 2 shows an overall picture of the normative framework.

Jess includes the possibility of using frame-based approaches, which should allow us to simplify the aggregation of contextual information and their usage

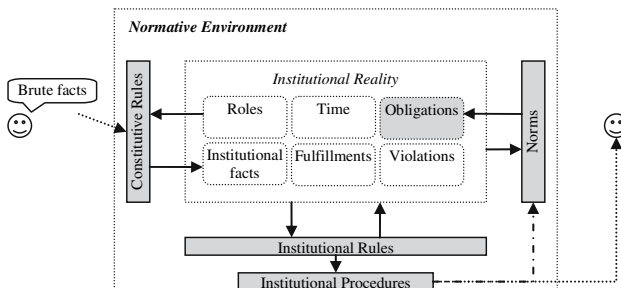


Fig. 2 Normative environment

within norms. Jess also includes the possibility of grouping norms into modules, which conceptually match our contexts.

6 Conclusions and related work

We have presented an electronic institution framework that incorporates a computable normative environment. In our approach, the EI is seen as a means to facilitate both the creation and the enforcement of contracts. Therefore, the EI acts as a trustable third-party providing an open environment with a flexible normative structure. Contractual norms are used to represent commitments that agents may create at run-time. Norms are kept in a declarative representation, and prescribed obligations have time references that the institution examines to trigger appropriate time ticks that allow for monitoring.

An advantage of a declarative normative framework is that agents can have access to their applicable norms, and can use what-if analysis to predict and reason about possible outcomes of both their and other agents' actions. This may be important in contracting situations, allowing potential contract breaches to be detected and prevented, or their damages minimized. It also allows the development of tools that make contract execution more prompt and collaborative (as explored in Sallé 2002).

Other approaches to the EI concept have considered it as a constraining infrastructure. In (Esteva et al. 2002), the EI is formally defined as a dialogical framework having a rigid performative structure (consisting of scenes and transitions among these scenes) that guides agent interactions. Complementing that approach, a middleware was developed (Esteva et al. 2004) which ensures that agents obey to the predefined protocol by constraining their allowed illocutions. Norms are therefore addressed as constraints on behavior, and cannot deal with situations where time is an issue. Nonetheless, their work (which builds on Rodríguez-Aguilar 2001) is arguably the most comprehensive one on the design, specification and development of electronic institutions (Arcos et al. 2005).

In our approach, we avoid imposing hard constraints on behavior. By enforcing norms (some of which may correspond to sanctions, that is, contrary-to-duty obligations (Jones and Carmo 2001)), we do conduct and supervise the behavior of rational agents.

Norms are typically related with the deontic notions of obligation, permission and prohibition, and have been used to formalize contracts (Sallé 2002; Kollingbaum and Norman 2002), or to specify interaction in agent societies (Dignum et al. 2003; Pacheco and Carmo 2003). In our case, we essentially rely on directed obligations to specify contractual commitments. Constitutive rules, by defining “counts-as” relations, specify empowerments of agents performing specific roles (Fornara et al. 2005; Jones and Sergot 1996). Given the institutional nature of such roles, for now we assume that agents are permitted to make their authorized statements. Contractual permissions are dealt with in an implicit way; norms can be triggered with institutional facts, which therefore can be used to exercise a right of demanding a certain contribution to a contractual partner. As for prohibitions, they

may be applied as a consequence of violation detection through institutional procedures. We leave for future work the explicit definition of contractual permissions and prohibitions, and the handling of their violation. However, in these cases violation detection is harder, because it assumes a pervasive character of the institution; actions may be performed which are not observable by the enforcing entity (Vázquez-Salceda et al. (2004) study different verifiability levels). As mentioned before, in respect to obligations we assume that it is in the best interest of contractual partners to inform the institution of their abidance to obligations. A formal underpinning of a logic for contract representation is given in (Dignum et al. 2003), including the notion of conditional obligation with deadline (equivalent to our norm specification).

Our norm formalism allows us to implement norm monitoring as a context-independent activity. Specifically, we distinguish violation detection from sanction imposition mechanisms. While the detection of violations is a general and institutionally defined concept, the prescription of sanctions may be contract-specific. This approach differs from (Vázquez-Salceda et al. 2004), where norms are defined as having explicitly associated violation conditions, means of detecting those violations, and sanction and repair measures. Therefore, their norm representation has a heavy structure, preventing the use of inheritance mechanisms that make the normative framework more flexible. In (García-Camino and Rodríguez-Aguilar 2005) the authors also follow a per-norm definition of violation detection rules, which translates in their Jess implementation to tailored rules that are added to verify norm compliance at the time when obligations are due.

López y López and Luck (2003) present a theoretical norm schema that includes, besides normative goals, preconditions and addressees, also rewards and punishments. While in order to be effective norms do need such elements (e.g., an obligation needs punishments), within a normative framework rewards and punishments are enforcement measures that may be institutionally defined. Again, we adopt a simpler and more modular norm specification, keeping sanction measures apart of each norm definition. The specification of norms in a contractual situation often configures a chaining of obligations, which resembles the notion of *interlocking norms* in López y López and Luck (2003): norms get activated through the fulfillment or un-fulfillment of other norms.

Artikis et al. (2002) developed a framework where an institution is seen as an external entity ascribing institutional powers, normative positions and sanctions to agents. Valid actions are distinguished from permitted ones, and the institution maintains a global state of the system, by checking norm compliance and employing sanctions (e.g., removing the power to execute an action) and enforcement policies. Our approach integrates some of these concepts, but tries to enrich the institution with a more supportive normative framework.

Searle's work on speech acts (Searle 1969) and institutional reality (Searle 1995) has inspired several researchers within the MAS field (e.g., Boella and van der Torre 2004; Colombetti and Verdicchio 2002; Fornara et al. 2005). In our case, we used this inspiration as a means to certify the occurrence of real-world actions, essential for contract monitoring purposes.

Colombetti and Verdicchio (2002) propose a distinction between natural actions (those that concern the activity of an agent in a physical environment) and institutional actions (which include performing speech acts). They define secondary actions as events intentionally brought about through the execution of another action, and relate this concept to the “counts as” relation of Searle. An institution defines contextual conditions for the application of instances of this relation. Fornara et al. (2005) define an artificial institution as an abstract description of shared concepts and rules that regulate agent interactions. Agents perform actions in a virtual environment (by exchanging messages) and the effects of such actions may also affect human reality. An artificial institution is composed of a core ontology (institutional concepts and actions that operate on them), a set of authorizations (empowerments of agents to perform institutional actions), and a set of norms (imposing obligations and permissions on the agents). Our approach includes these concepts, but also assigns to the institution a set of services that are meant to assist (not only regulate) agent interaction and the creation of new normative relationships. Another similarity of our approach with that of Fornara et al. is their view of norms as rules that manipulate agent commitments, and that strictly speaking norms cannot be fulfilled or violated; instead, what is subject to violation is the commitments created by the application of a norm (e.g., obligations).

Our approach also tries to incorporate some concepts of real-world contracting. The hierarchical normative structure allows us to define “default rules” (Craswell 2000) that are to be inherited by certain types of contracts. These rules (which are in fact norms in our case) form the body of law that regulates in a standard way a certain activity. The most useful case for this is in defining contrary-to-duty situations, which typically should be not likely to occur. For this reason, such situations are not dealt with in each contractual agreement, and agents usually recur to legislative systems that define default procedures (Daskalopulu and Maibaum 2001).

We are in the process of developing a prototype that includes all the concepts presented in this paper. Work is also being devoted to study the applicability of our approach in VO scenarios closer to real-world situations, in order to get example illustrations that allow us to show the usefulness of the prototype.

Acknowledgment The work reported in this paper is supported by the FCT (Fundação para a Ciência e a Tecnologia) Project POSC/EIA/57672/2004. Henrique Lopes Cardoso enjoys the FCT grant SFRH/BD/29773/2006.

References

- Arcos JL, Esteva M, Noriega P, Rodríguez-Aguilar JA, Sierra C (2005) Environment engineering for multiagent systems. *Eng Appl Artif Intel* 18(2):191–204
- Artikis A, Pitt J, Sergot M (2002) Animated specifications of computational societies. In: *Proceedings 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'02)*, Bologna, July 2002, pages 1053–1062. ACM Press
- Boella G, van der Torre L (2004) Regulative and constitutive norms in normative multiagent systems. In: Dubois D, Welty C, Williams M-A (eds) *Proc. 9th Int. Conf. principles of knowledge representation and reasoning*. Canada, Whistler, pp 255–266

- Boella G, van der Torre L, Verhagen H (2005) Introduction to normative multiagent systems in 1st Int. Symposium on Normative Multiagent Systems. England, Hatfield
- Castelfranchi C (2000) Engineering social order. In: Omicini A, Tolksdorf R, Zambonelli F (eds) *Engineering societies in the agents world*. Springer, pp 1–18
- Castelfranchi C, Dignum F, Jonker C, Treur J (2000). Deliberative normative agents: principles and architectures. In: Jennings N, Lesperance Y (eds) *Intelligent agents VI: agent theories, architectures, and languages*. Springer, pp 364–378
- Colombetti M, Verdicchio M (2002) An analysis of agent speech acts as institutional actions. In: Proc. 1st Int. Joint Conf. on autonomous agents and multiagent systems. ACM Press, New York, pp 1157–1164
- Conte R, Falcone R, Sartor G (1999) Introduction: agents and norms: how to fill the gap? *Artif Intel Law* 7(1):1–15
- Craswell R (2000) Contract law: general theories. In: Bouckaert B, De Geest G (eds) *Encyclopedia of law and economics, vol III: the regulation of contracts*. Edward Elgar, Cheltenham, pp 1–24
- Daskalopulu A, Maibaum T (2001) Towards Electronic Contract Performance. *Legal Information Systems Applications, 12th International Conference and Workshop on Database and Expert Systems Applications*. IEEE Computer Society, pp 771–777
- Dignum V, Dignum F (2002) Towards an agent-based infrastructure to support virtual organizations. In: Camarinha-Matos LM (ed) *Collaborative business ecosystems and virtual enterprises*. Kluwer, pp 363–370
- Dignum V, Meyer J-J, Dignum F, Weigand H (2003) Formal specification of interaction in agent societies. In: Hinchey M, Rash J, Truszkowski W, Rouff C, Gordon-Spears D (eds) *Formal approaches to agent-based systems*. Springer, pp 37–52
- Esteva M, Padget J, Sierra C (2002) Formalizing a language for institutions and norms. In: Meyer J-J, Tambe M (eds) *Intelligent agents VIII*. Springer, pp 348–366
- Esteva M, Rosell B, Rodríguez-Aguilar JA, Arcos JL (2004) AMELI: an agent-based middleware for electronic institutions. In: Proc. 3rd Int. Joint Conf. on Autonomous Agents and Multi-agent Systems: vol 1. IEEE Computer Society, New York, pp 236–243
- FIPA (2002a) FIPA-ACL Message Structure Specification, <http://www.fipa.org/specs/fipa00061/>
- FIPA (2002b) FIPA Communicative Act Library Specification, <http://www.fipa.org/specs/fipa00037/>
- FIPA (2002c). FIPA Agent Message Transport Service Specification, <http://www.fipa.org/specs/fipa00067/>
- Fornara N, Viganò F, Colombetti M (2005) Agent communication and institutional reality. In: van Eijk RM, Huet M-P, Dignum F (eds) *Agent communication: international workshop on agent communication, Lecture Notes in Computer Science, vol 3396*. Springer, pp 1–17
- Friedman-Hill E (2003) *Jess in action*. Manning Publications Co
- García-Camino A, Rodríguez-Aguilar JA (2005) Implementing norms in electronic institutions. In: Proc. 4th Int. Joint Conf. on Autonomous Agents and Multi-agent Systems. ACM Press, New York, pp 667–673
- Jones A, Carmo J (2001) Deontic logic and contrary-to-duties. In: Gabbay D (ed) *Handbook of philosophical logic*. Kluwer, pp 203–279
- Jones A, Sergot M (1996) A formal characterisation of institutionalised power. *Logic J IGPL* 4(3):427–443
- Kollingbaum MJ, Norman TJ (2002) Supervised interaction—creating a web of trust for contracting agents in electronic environments. In: Proc. 1st Int. Joint Conf. on Autonomous Agents and Multi-agent Systems. ACM Press, New York, pp 272–279
- López y López F, Luck M (2003) Modelling norms for autonomous agents. In: Chavez E, Favela J, Mejia M, Oliart A (eds) *Proc. 4th Mexican Int. Conf. on Computer Science, IEEE Computer Society*, pp 238–245
- Lopes Cardoso H, Malucelli A, Rocha AP, Oliveira E (2005) Institutional services for dynamic virtual organizations. In: Camarinha-Matos LM, Afsarmanesh H, Ortiz A (eds) *Collaborative networks and their breeding environments—6th IFIP working conference on virtual enterprises (PRO-VE'05)*. Springer, pp 521–528
- Lopes Cardoso H, Oliveira E (2005a) Assisting and regulating virtual enterprise interoperability through contracts. In: Berre A, Elms K, Fischer K, Mueller JP (eds) *Proc. Agent-based Technologies and applications for enterprise interOPERability*. The Netherlands, Utrecht

- Lopes Cardoso H, Oliveira E (2005b) Virtual enterprise normative framework within electronic institutions. In: Gleizes M-P, Omicini A, Zambonelli F (eds) *Engineering societies in the agents World V*. Springer, pp 14–32
- Malucelli A, Palzer D, Oliveira E (2005) Combining ontologies and agents to help in solving the heterogeneity problem. In: *Proceedings of the international workshop on data engineering issues in e-commerce (DEEC2005)*. IEEE Computer Society, Tokyo, Japan, pp. 26–35
- Oliveira E, Rocha AP (2000) Agents advanced features for negotiation in electronic commerce and virtual organisations formation process. In: Dignum F, Sierra C (eds) *Agent mediated electronic commerce: the european agentlink perspective*. Springer, pp 78–97
- Pacheco O, Carmo J (2003) A role based model for the normative specification of organized collective agency and agents interaction. *J Autonomous Agents Multi-Agent Syst* 6(2):145–184
- Rodríguez-Aguilar JA (2001) *On the design and construction of Agent-mediated Electronic Institutions*, Ph.D. Thesis, Universitat Autònoma de Barcelona
- Sallé M (2002) Electronic contract framework for contractual agents. In: Cohen R, Spencer B (eds) *Advances in artificial intelligence: 15th conference of the Canadian society for computational studies of intelligence*. Springer, pp 349–353
- Searle JR (1969) *Speech acts: an essay in the philosophy of language*. Cambridge University Press, Cambridge
- Searle JR (1995) *The construction of social reality*. Free Press, New York
- Vázquez-Salceda J, Aldewereld H, Dignum F (2004) Implementing norms in multiagent systems. In: Lindemann G, enzinger J, Timm IJ, Unland R (eds) *Multiagent system technologies*. Springer, pp 313–327