Auton Agent Multi-Agent Syst (2007) 14:121–142
DOI 10.1007/s10458-006-0017-8

Agent communication and artificial institutions

Nicoletta Fornara · Francesco Viganò · Marco Colombetti

Published online: 31 August 2006
Springer Science+Business Media, LLC 2006

Abstract In this paper we propose an application-independent model for the definition of artificial institutions that can be used to define open multi-agent systems. Such a model of institutional reality makes us able also to define an objective and external semantics of a commitment-based Agent Communication Language (ACL). In particular we propose to regard an ACL as a set of conventions to act on a fragment of institutional reality, defined in the context of an artificial institution. Another contribution of the work presented in this paper is an operational definition of norms, a crucial component of artificial institutions. In fact in open systems interacting agents might not conform to the specifications. We regard norms as event-driven rules that when are fired by events happening in the system create or cancel a set of commitments. An interesting aspect of our proposal is that both the definition of the ACL and the definition of norms are based on the same notion of commitment. Therefore an agent capable of reasoning on commitments can reason on the semantics of communicative acts and on the system of norms.

Keywords Agent communication language · Commitments · Artificial institutions · Norms

N. Fornara (✉) · F. Viganò · M. Colombetti
Università della Svizzera italiana, via G. Buffi 13, 6900 Lugano, Switzerland
e-mail: nicoletta.fornara@lu.unisi.ch

F. Viganò
e-mail: francesco.vigano@lu.unisi.ch

M. Colombetti
Politecnico di Milano, piazza Leonardo Da Vinci 32, Milano, Italy
e-mail: marco.colombetti@lu.unisi.ch

1 Introduction

In this paper we propose an application-independent model of institutional reality that can be used to define open multi-agent systems as the reification of a set of *artificial institutions* and that makes us able to define the semantics of a set of communicative acts suitable for agent communication.

Our approach to the definition of communicative acts is based on the notion of *social commitment*. In the last few years this concept has been largely used by a growing number of researchers to define the semantics of Agent Communication Languages (ACLs). After the first studies carried out by Singh and by Colombetti [34, 8], further investigations have been carried out from an operational point of view [17, 27], following a logical approach [36], and in the field of argumentation studies [1, 5]. The main advantages of this approach are that commitments are objective and independent of an agent's internal structure and that it is possible to verify whether an agent is behaving according to the given semantics. Therefore the proposed ACL, unlike the ACLs that define the meaning of communicative acts using agent's mental states [21, 14], is suitable to be used in open systems where no constraints are imposed on the internal way of reasoning of the interacting agents and where it is impossible to trust other agents completely.

Social commitments are used to represent the evolution of social relationships among agents during interactions. Communicative acts are then viewed as actions carried out to modify social relationships among agents by creating, updating or cancelling commitments according to a predefined set of shared rules [36, 17]. More precisely, we regard communicative acts as *institutional actions*, that is, as actions performed within an institution to modify a fragment of social reality [31]. Defining the semantics of an ACL has therefore two sides: one side is the definition of the institutional effects brought about by the performance of communicative acts; the other side is the definition of the social context in which agents can carry out institutional actions, and that we call artificial institutions. Indeed, our main tenet is that without the definition of appropriate institutions it is impossible to specify the semantics of an ACL.

An important component of our model of artificial institutions is the definition of *norms*. In fact in open systems there is the risk that the agents do not conform to the specifications. We regard norms as event-driven rules that, when are fired by events happening in the system, create or cancel a set of commitments. Therefore a crucial aspect of the proposed model is that the definition of the ACL and the definition of norms are based on the same concept: the notion of commitment. As a consequence an agent capable of reasoning on commitments can reason on the semantics of communicative acts and on the system of norms.

The paper is organized as follows. In Sect. 2 we describe our model of artificial institutions that consists of the fundamental concepts on which we base our treatment of agent communication. In Sect. 3 we introduce the basic institution, which defines the notion of social commitment as an institutional entity and which regulates their management. In Sect. 4 we consider a further component of artificial institution, namely norms. We then give two examples of applications of our model: in Sect. 5 we delineate a FIPA-like library of communicative acts by defining a set of institutional actions and suitable conventions for their performance and in Sect. 6 we give a partial description of a specific artificial institution, that is, the institution of English Auction

(EA). In Sect. 7 we briefly remark on related works present in this volume. Finally in Sect. 8 we draw some conclusions and delineate some directions for future work.

2 Artificial institutions

We view a multiagent system (MAS) as a technological extension of human society, by which individual persons and human organizations can delegate the execution of institutional actions to an artificial system. Examples of such actions are establishing appointments, signing contracts, and carrying out commercial transactions. For this reason there are strong connections between some aspects of a MAS and some aspects of human society, and therefore the concepts used to model a MAS interaction framework have to reflect some crucial characteristics of their human counterparts. For this reason in creating our model of artificial institutions we draw inspiration from philosophical studies about social reality and human communication [30, 31].

The word *institution* is normally used with different meanings. An institution can be seen as an established organization (especially of a public character) with a code of law, like for example a hospital or a university. With a different meaning, the word is used to refer to a set of concepts that exist only thanks to the common agreement of a community of agents, like for example in the case of money, ownership, or marriage.

In multiagent systems research the term *electronic institution* is commonly used to refer to a specific organization [12] or to an abstract pattern that regulates the interaction among agents [35]. On the contrary, we use the term "artificial institution" to refer to the description of shared concepts and rules that create and regulate a fragment of social reality. In this perspective a concrete open MAS is a reification of one or more artificial institutions.

In our view, the specification of an artificial institution consists of the following components:

– The *core ontology*, that is, the definition of the *institutional concepts* introduced by the institution and of the *institutional actions* that operate on them;
– a set of *authorizations* specifying which agents are empowered to perform institutional actions;
– a set of *conventions* for the concrete performance of institutional actions;
– a set of *norms* that impose *obligations* and *prohibitions* on the agents that interact within the MAS.

Of course, in order that the proposed model can be used in real applications it is necessary that these fundamental concepts, used to define the structure of institutions, are collectively accepted by the designers of open interaction frameworks and by the designers of artificial agents that will interact with that framework.

2.1 The core ontology: entities and attributes

The context within which artificial agents operate can be modeled as consisting of a set of *entities* that can have *natural* or *institutional* attributes, that is, attributes that exist only thanks to the common agreement of the interacting agents (or more precisely of their users). For example, the color of a book is a natural attribute, while the book's price is an institutional attribute. Natural attributes are assumed to reflect the physical properties of the corresponding entities of the real world, and typically cannot be changed by artificial agents (unless the agent controls a physical robot). On the contrary, institutional attributes can be affected by *institutional actions* performed by software agents.

Sometimes, core ontologies define entities whose attributes are only institutional. We refer to such entities as *institutional entities*. As will be described in Sect. 3, social commitments are the fundamental institutional entities, because they are essential to express the meaning of various communicative acts.

2.2 Institutional actions

Institutional actions are a particular type of actions [10] that are crucial for the formalization of communicative interactions taking place in open systems. The effect of institutional actions is to change institutional attributes, which exist only thanks to common agreement. Therefore, agents cannot perform such actions by exploiting causal links occurring in the natural world, as would be done to open a door or to remove a physical object. Rather, as we shall see, institutional actions are performed on the basis of a particular construct: the "*counts-as*" relation.

Because of their intrinsic social nature, a crucial condition for the actual performance of institutional actions is that they are *public*, that is, made known to the relevant agents by means of some action that can be directly executed by an artificial agent. It is therefore natural to assume that all institutional actions are performed by sending suitable messages to the relevant agents. An example of institutional action is the act of opening an auction; as we shall see, an agent (the auctioneer) can perform such an action by sending a suitable message to the relevant group of agents (the participants). However, the act of sending the message is merely instrumental, and should not be confused with the institutional action of opening the auction.

In the literature, mechanisms different from message exchange have also been considered. For example, the use of blackboards for MAS has been proposed for regulating interaction in open systems by several authors, i.e. [38]. It is worth noticing, however, that FIPA standards see agents as entities that communicate only by ACL messages. In any case, we think that the change of an institutional attribute, like the one representing that an auction is open, always corresponds to the execution of a speech act, namely, a declaration. When agents use a blackboard, the declaration is performed by writing a suitable data structure on the blackboard, and no message is exchanged among the agents. But still, the institutional action is performed by executing a lower-level, conventional action (writing on a blackboard instead of exchanging a message).

We define institutional actions by specifying their *preconditions* and *postconditions*, therefore abstracting from the way in which such actions are concretely carried out. More precisely, an institutional action is characterized by:

– An *action name* followed by a possibly empty list of *parameters*;
– a possibly empty set of *preconditions*, which specify the values that certain institutional attributes must have for the action to be meaningful (for example, opening an auction is meaningful only if the auction is not already open);
– a nonempty set of *postconditions*, which typically represent the effects of the action, that is, specify the values of certain institutional attributes after the performance of the action (for example an auction is necessarily open after opening it).

As we will see later in Sect. 5, communicative acts are a particular type of institutional actions.

2.3 Instrumental actions

As we have already remarked, an institutional action is performed by executing an instrumental action conventionally associated to the institutional action. In the human world such instrumental actions vary from certain bodily movements (raising one's arm to vote), to the use of specific physical tools (waving a white flag to surrender), to the use of language (saying "the auction is open" to open an auction). In a system of artificial agents, it is natural to assume that all institutional actions are performed by means of a single type of instrumental actions, namely exchanging a message.

For the purposes of the current treatment, a message consists of: a *message type*, a *sender*, one or more *receivers*, and a *content*. The action of exchanging a message will be represented with the following notation:

$$exchMsg(message\_type, sender, receiver(s), content)$$

Note that here *sender* and *receiver* are just fields of a message. That such fields correctly represent the agent that actually sends the message and the agents to which the message is delivered has to be guaranteed by the underlying message transport system.

2.4 The "counts-as" relation

What is the relation that binds the performance of an instrumental action to the performance of an institutional action? Following Searle [31], the construction of social reality in the human world is possible thanks to *constitutive rules* of the form *X counts as Y in C*. When X and Y are actions, such constitutive rules make it the case that performing an action of type X in context C counts as performing an action of type Y. In an artificial system the "counts-as" relation can be used to bind the performance of a message exchange to the corresponding institutional action, if certain contextual conditions are satisfied. In such a case, constitutive rules of the form *X counts as Y in C* represent linguistic conventions, which associate an institutional effect (Y) to the performance of a message exchange (X) in a suitable context (C). In this paper, the treatment of the counts-as relation is only concerned with such linguistic conventions. Thefore we do not deal with other aspects of counts-as, like for example classification in context, which in the area of artificial institutions has been proposed as a mechanism for the refinement of abstract norms [22].

2.4.1 Conventions

In order to be able to model the connection between X and Y we introduce the notion of *convention*, that is, an agreement about what type of message is bound to a given type of institutional action. In our model the definition of a convention has the following generic form:

$$exchMsg(message\ type, sender, receivers, content) =_{conv} iaction(parameters)$$

In Sect. 5 some useful conventions will be defined for the performance of institutional actions (in particular communicative acts).

2.4.2 Contextual conditions

By itself, a convention is not sufficient to guarantee the successful performance of an institutional action by the exchange of the appropriate message: indeed, some additional conditions about the agent that sends the message, about the agents that receive the message, and about the state of the system in relation to the content of the message must be satisfied.

Conditions on the sender of the message: In general, an agent must be *authorized* to perform an institutional action; for example, only the auctioneer has the power to open an auction by sending a suitable message to the participants. Moreover an authorization can be given only if certain conditions about the state of the system, expressed by suitable Boolean expressions, are satisfied. For example, it may be established that an auction is validly opened only if there are at least two participants. Assuming that every agent in the interaction system has an identifier (*agent_id*), authorizations will be represented with the following notation:

$$Auth(agent\_id, iaction(parameters), conditions)$$

Our notion of authorization should not be confused with the notion of permission. The distinction we make between these two concepts is similar to the one between institutionalized power and permission proposed by Jones and Sergot [23] and further discussed in [3]. While authorizations are necessary conditions for the valid performance of institutional actions, permissions are brought about by the absence of norms (see Sect. 4), that is, by rules that affect the normative positions of the agents in the system. The crucial difference between authorizations and permissions is highlighted in the cases when they are not granted. If an agent is not authorized to perform an institutional action, a performance of the corresponding instrumental action does not count as a performance of the institutional action (the institutional action is thus *not executed*). On the contrary, if an authorized agent performs an institutional action that is prohibited by a norm, the institutional action is successfully performed, but the agent violates the norm and may be sanctioned for its behavior.

In the specification of an interaction system it is useful to express authorizations in term of the *roles* filled by agents, in order to abstract from the concrete agents that are actually involved in an interaction. For example, the authorization to open and close an auction is granted to the agent that fills the role of the auctioneer, independently of its individual identity. We can then abstractly define the authorization to perform a specific institutional action (with given parameters) associating it to a role:

$$Auth(role, iaction(parameters), conditions)$$

In a concrete interaction, starting from the authorizations given in terms of roles and reasoning on agent roles, it is possible to detect if a given agent is authorized to perform a certain institutional action.

Conditions about the receivers of the message A message must be received by all agents that play a role in the institutional reality affected by the performance of the act. For example if the institutional action operates on a commitment, its debtor and/or its creditor (depending on who is the actor of the action) have to receive the message; in the case that the institutional action opens an auction, the participants in the auction have to be the receivers of the message.

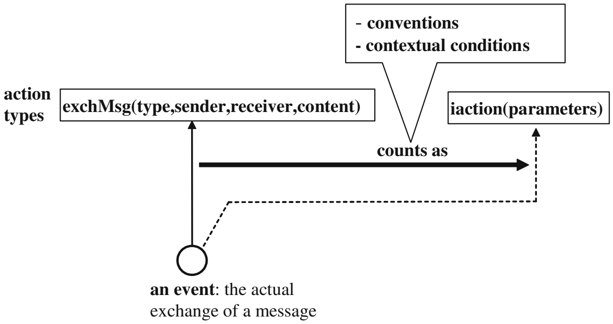
- conventions
- contextual conditions
action types
exchMsg(type,sender,receiver,content)
iaction(parameters)
counts as
an event: the actual exchange of a message

Fig. 1 The "counts-as" relation

Conditions about the state of the system All the preconditions of the institutional action associated to the performance of the exchange of the message must be satisfied.

Figure 1 schematically depicts how the "counts-as" relation works: an actual exchange of a message counts as the performance of a particular institutional action if there is a convention that binds the two acts and the relevant contextual conditions are satisfied.

3 The Basic Institution

In [16–18] we have proposed to define the semantics of ACLs starting from the assumption that the performance of certain communicative acts in a open multiagent system has the effect of changing the social relationship between the sender and the receivers, and that this change can be represented by means of an institutional entity, that is, *social commitment*. To specify the meaning of various types of communicative acts in terms of effects on commitments, it is necessary to define an ontology of commitment, the institutional actions necessary to operate on commitments, and a set of authorizations for the performance of these institutional actions. As we shall see, all institutional actions on commitments can be performed by means of a basic communicative act: the *declaration*. In Sect. 5 we will define a set of conventions necessary to perform some type of communicative acts directly by the exchange of a suitable message.

Commitment entities are fundamental not only for the definition of the semantics of an ACL but also for the definition of the semantics of norms as we will see in Sect. 4. Therefore in this section, using the concepts defined so far for describing the institutions that characterize an open interaction framework, we will define the *basic institution*, that is, the institution that defines and regulates the management of commitments.

Further institutions, that we call *special institutions*, can then be defined to model the aspects of institutional reality typical of certain application domains. For instance, for electronic commerce applications it will be necessary to model the institutions of ownership, money, business transactions, auctions, and so on. A formal specification of the EA as a special institution is given in Sect. 6.

3.1 The ontology of commitment

We regard a commitment as an institutional entity with the following attributes: a *debtor*, a *creditor*, a *content*, and a *state* which is used to keep track of the temporal evolution of the commitment. We assume that given a debtor, a creditor, and a content a commitment could be univocally identified. Commitments will be represented with the following notation:

$$Comm(state, debtor, creditor, content)$$

The content of a commitment is represented by means of a *temporal proposition* (for a detailed treatment of temporal propositions (see [17–9]) that relates a state of affairs or the performance of an action to an interval of time. Temporal propositions are characterized by a *statement* having the following *expressionType*: *action*, *proposition*, or *referential expression*, which is referred to a specific *interval of time* with two different modes: ∀ and ∃. It is up to a "notifier", connected with the meaning of the statement, to change the *state* from *undefined*($\perp$), to *true*, or *false*. If the *mode* is ∃ and the *statement* becomes true at any point of the *time interval* the notifier sets the *state* to true; otherwise the notifier sets the *state* to false when the *time interval* expires. If the *statement* is negated, the notifier sets the *state* to false when the *statement* becomes true, otherwise it sets the state to true at the end of the *time interval*. The mode ∀ is treated as the dual of ∃. Temporal propositions are represented with the following notation:

$$\mathrm{TP}(statement, [t_{\mathrm{start}}, t_{\mathrm{end}}], mode, state)$$

For example the following temporal proposition (used in Sect. 6.3) can be used to monitor that a certain auction, identified by an *id* is actually opened between *now* and $now + \delta$ is:

$$\mathrm{TP}(openAuction(id), [now, now + \delta], \exists, \perp)$$

We perceive that commitments to actions and commitments to propositions have different aspects [39], but a detailed treatment of these is beyond the scope of this paper.

The state of a commitment undergoes the life cycle described by the state diagram of Fig. 2, and changes as an effect of the execution of institutional actions (solid lines) or of environmental events (dotted lines). Relevant events are due to the change of the truth-value of the commitment's content. In particular by definition a commitment is *fulfilled* when its content becomes true and is *violated* when its content becomes false.

The creditor of a commitment can be a single agent or a group of agents. It is important to remark that a commitment taken with a group of agents need not be equivalent to a conjunction of commitments taken with every member of the group. This point has been thoroughly analyzed in the literature [7, 11] and will not be dealt here.

In our approach commitments can be created by individual agents through the execution of communicative acts or by the activation of norms and which generate commitments for agents in virtue of their role in an institution.

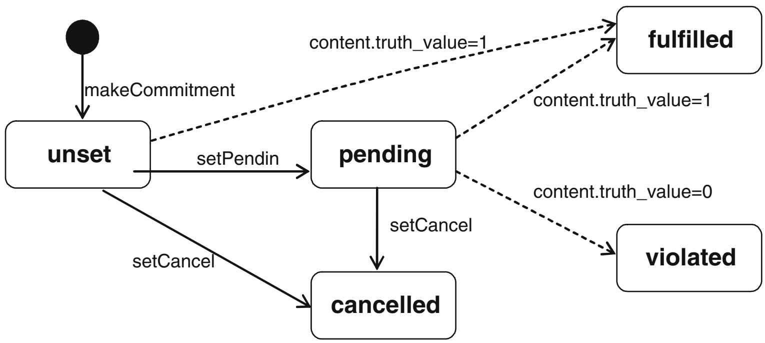

Fig. 2 The life-cycle of commitments

3.1.1 Institutional actions on commitment

The institutional actions that operate on commitments are defined below; preconditions and post-conditions are described using Object Constraint Language (OCL) [29]. The *makeCommitment* action creates a new *unset* commitment if a commitment with the same debtor, creditor, and content does not exist yet:

$$
\begin{aligned}
&\mathit{name}: \boldsymbol{makeCommitment}(\mathit{debtor}, \mathit{creditor}, \mathit{content})\\
&\mathit{pre}: \quad \mathit{not\ Comm.allInstances}() \rightarrow \mathit{exists}(c|c.\mathit{debtor} = \mathit{debtor}\\
&\qquad \mathit{and\ c.creditor} = \mathit{creditor\ and\ c.content} = \mathit{content})\\
&\mathit{post}: \quad \mathit{Comm.allInstances}() \rightarrow \mathit{exists}(c|c.\mathit{state} = \mathit{unset\ and\ c.debtor}\\
&\qquad = \mathit{debtor\ and\ c.creditor} = \mathit{creditor\ and\ c.content} = \mathit{content})
\end{aligned}
$$

The *setCancel* action changes the state of an existing *unset* or *pending* commitment to *cancelled*:

$$
\begin{aligned}
&\mathit{name} := \boldsymbol{setCancel}(\mathit{debtor}, \mathit{creditor}, \mathit{content})\\
&\mathit{pre}: \quad \mathit{Comm.allInstances}() \rightarrow \mathit{exists}(c|(c.\mathit{state} = \mathit{unset\ or}\\
&\qquad c.\mathit{state} = \mathit{pending})\ \mathit{and\ c.debtor} = \mathit{debtor\ and\ c.creditor} =\\
&\qquad \mathit{creditor\ and\ c.content} = \mathit{content})\\
&\mathit{post}: \quad \mathit{Comm.allInstances}() \rightarrow \mathit{exists}(c|c.\mathit{state} = \mathit{cancelled\ and\ c.debtor}\\
&\qquad = \mathit{debtor\ and\ c.creditor} = \mathit{creditor\ and\ c.content} = \mathit{content})
\end{aligned}
$$

Finally the *setPending* action transforms an *unset* commitment to a *pending* one:

$$
\begin{aligned}
&\mathit{name}: \boldsymbol{setPending}(\mathit{debtor}, \mathit{creditor}, \mathit{content})\\
&\mathit{pre}: \quad \mathit{Comm.allInstances}() \rightarrow \mathit{exists}(c|c.\mathit{state} = \mathit{unset\ and\ c.debtor}\\
&\qquad = \mathit{debtor\ and\ c.creditor} = \mathit{creditor\ and\ c.content} = \mathit{content})\\
&\mathit{post}: \quad \mathit{Comm.allInstances}() \rightarrow \mathit{exists}(c|c.\mathit{state} = \mathit{pending\ and\ c.debtor}\\
&\qquad = \mathit{debtor\ and\ c.creditor} = \mathit{creditor\ and\ c.content} = \mathit{content})
\end{aligned}
$$

For convenience, we also define the action *makePendingCommitment*, which corresponds to the sequential execution of *makeCommitment* and *setPending*. *MakePendingCommitment* therefore creates a *pending* commitment if it is not the case that there exists a commitment with the same creditor, debtor, and content.

$$
\begin{aligned}
&\mathit{name}: \boldsymbol{makePendingCommitment}(\mathit{debtor}, \mathit{creditor}, \mathit{content})\\
&\mathit{pre}: \quad \mathit{not\ Comm.allInstances}() \rightarrow \mathit{exists}(c|c.\mathit{debtor} = \mathit{debtor}
\end{aligned}
$$

$$\begin{aligned} &\quad \textit{and c.creditor} = \textit{creditor and c.content} = \textit{content})\\ \textit{post}:\ &\textit{Comm.allInstances}() \rightarrow \textit{exists}(c|\textit{c.state} = \textit{pending and c.debtor}\\ &= \textit{debtor and c.creditor} = \textit{creditor and c.content} = \textit{content}) \end{aligned}$$

3.2 Authorizations

We define a set of authorizations concerning the creation and the manipulation of commitments. Such authorizations will be associated with the two roles introduced by commitments themselves: the role of *debtor* and the role of *creditor*. Moreover, we assume a universal role, *RegAgt*, that every registered agent plays throughout its lifetime.

(1) Any registered agent can create an *unset* commitment with any other registered agent as debtor or creditor:

$\textit{Auth}(\textit{RegAgt}, \textbf{makeCommitment}(\textit{debtor}, \textit{creditor}, \textit{content}), \textit{true});$

(2) the debtor of an *unset* commitment can set it to *pending*:

$\textit{Auth}(\textit{Comm}(\textit{debtor}, \textit{creditor}, \textit{content}).\textit{debtor},$
$\quad \textbf{setPending}(\textit{debtor}, \textit{creditor}, \textit{content}), \textit{true});$

(3) the debtor of an *unset* commitment can set it to *cancelled*:

$\textit{Auth}(\textit{Comm}(\textit{unset}, \textit{debtor}, \textit{creditor}, \textit{content}).\textit{debtor},$
$\quad \textbf{setCancel}(\textit{debtor}, \textit{creditor}, \textit{content}), \textit{true});$

(4) the creditor of a commitment can set it to *cancelled*:

$\textit{Auth}(\textit{Comm}(\textit{debtor}, \textit{creditor}, \textit{content}).\textit{creditor},$
$\quad \textbf{setCancel}(\textit{debtor}, \textit{creditor}, \textit{content}), \textit{true});$

(5) since a registered agent can create an *unset* commitment with itself as debtor and any other agent as creditor, and the debtor of an *unset* commitment can set it to *pending*, a registered agent is authorized to create a *pending* commitment with itself as debtor and any other agent as creditor:

$\textit{Auth}(\textit{RegAgt}, \textbf{makePendingCommitment}(\textit{RegAgt}, \textit{creditor}, \textit{content}), \textit{true}).$

Notice that these authorizations allow an agent to perform all communicative acts defined in Sect. 5. New authorizations may be introduced within special institutions.

In general, institutions also define sets of norms to regulate the behavior of agents. In our current view, the basic institution does not specify norms. However, norms are introduced by most special institutions, as exemplified by the special institution of EA described in Sect. 6.

4 Norms

In a special institution, the execution of an action by an authorized agent often needs to be regulated by another fundamental component of artificial institutions, that is, a system of *norms*. For example, the auctioneer of an EA not only is authorized to declare who is the winner, but it is also obliged to do so in certain circumstances.

Furthermore, there are conditions under which it is forbidden to the auctioneer to declare an agent as the winner (for instance during a period of time reserved for offers).

Norms prescribe which institutional actions should or should not be executed among those that are authorized. In doing so, norms play an important function, in that when they are respected norms make an agent's behavior at least partially predictable and allow agents to coordinate and plan their actions according to the expected behavior of the others, as studied in [28, 4]. We think that norms can also be used to specify protocols, because they can dictate that in certain circumstances an agent should not perform several kinds of communicative acts, it ought to send a given type of message or react to a message in a specific way. How this can be done will be shown in Sect. 6.

We regard norms as event-driven rules that fire under appropriate conditions and, by doing so, create, update or cancel commitments affecting a predefined set of agents. To define our model of norms and how they react to events, in [19] we have proposed to model type of events as stereotyped classes [6] having attributes that provide information about the state transition that caused them. In our formalization we have singled out three main categories of events:

(1) *Timeevent*s, which occur when the system reaches a certain instant of time.
(2) *Changeevent*s, which happen when an institutional entity changes in some way, for example when an attribute modifies its value.
(3) *Actionevent*s, which happen when an agent perform an action (an interesting type of this kind of events is *exchMsg* (see Sect. 2.3), which represents the act of sending a message).

The definition of event types allows us to describe *event templates* [20], that is, event types with some restrictions on certain attributes that describe a set of possible event occurrences.

A norm is defined within an institution and observes an entity of that institution by registering an event template to be notified whenever an event matching the template occurs. Typically, interesting event types are not only communicative acts like in [13], but also the filling of a role by an agent, a value change of an institutional attribute, the reaching of certain instants of time, and so on. When an event matches the given descriptor, the corresponding norm is fired, its variable *e* is filled with the event, and the norm is activated.

When a norm fires, it is applied to a collection of *liable agents* described by a suitable selection expression; in general, the collection of liable agents corresponds to the set of agents that play a given role in the institution. For every liable agent, the norm creates, updates or cancels a set of commitments.

In our model, when an agent fills a role in a software system implementing an institution, the norms of that institution will create commitments binding the agent to the system, which is itself regarded as an agent. The identity of the creditor agent allows us to keep trace of the commitments created by a system and thus, in case of need depending on the application domain, to distinguish them from the ones created by other agents.

The general structure of a norm can be described as follows:

within *context_name*: *ientity*
on *e*: *event_template*

```
if contextual conditions then
  foreach agent in liable agents
  do $\{commActionDescription(agent, system, parameters)\}^+$
```

Many studies have been devoted to the analysis of the relationship holding between norms and commitments, which is often perceived as a fundamental aspect of institutions [12] and organizations [7]. For example in [25] commitments are viewed as a specialization of norms, while in [7] and [33] norms are a special kind of commitments, called *metacommitments*.

From our point of view, norms are not themselves commitments, but rules that manipulate commitments of the agents engaged in an interaction. In fact, norms are associated to roles rather than to individual agents; they do not have a debtor or a creditor, and strictly speaking they cannot be fulfilled or violated. Indeed, what can be fulfilled or violated is not a norm, but commitments created by the application of a system of norms, which represent agents' *obligations* and *prohibitions*. More precisely, obligations are commitments to perform an action of a given type, and prohibitions are commitments not to perform an action of a given type. We interpret the absence of positive or negative commitments to the execution of an action of a given type as *permissions*. This means that in our framework every action is permitted unless it is explicitly forbidden. Other authors [2, 3] take a different standpoint, and assume that every action is prohibited unless it is explicitly permitted. We find this choice rather unnatural, because it makes it necessary to explicitly specify all actions that may be performed without incurring a violation.

Usually in the agent literature authorization is not distinguished from permission. For example, in [13] agent interactions are specified through finite state machines, which represent both authorized and permitted acts. The distinction between authorization and permission becomes fundamental when new institutions are defined by using previously defined institutions [37]. In fact, when a designer defines a new and independent institution, he or she can arbitrary decide to constrain agent behavior by limiting authorizations or by introducing new norms. Instead, as discussed in [37], when different institutions interact, institutional actions authorized to an agent by another institution can be further conditioned only through norms.

In the following two sections we will exemplify our model by defining a FIPA-like library of communicative acts and by formalizing the English Auction as an artificial institution.

5 A commitment-based agent communication language

In this section we define a library of communicative acts, a particular type of institutional actions, and the set of conventions necessary for their actual performance. To be compatible with the syntax of FIPA-ACL [21], when possible we name our communicative acts with the FIPA performative that has the closest intuitive meaning.

Content language expressions of communicative acts are *temporal propositions* having an *expressionType* that allows us to define conditions on the kind of expression that can be used as content of each message and as parameter of each communicative act (see Sect. 3).

In the sequel, the semantics of communicative acts is partly given in terms of preconditions and postconditions of other institutional action. In fact, most of the communicative acts, except for declarations, differ from institutional actions defined by the basic institution only because they have additional conditions on the type of the content. Thus, we introduce two new OCL operators, *precondition*, and *postcondition*, to evaluate the precondition and postcondition of an institutional action.

The act of informing, whose main point is to commit the sender of the message to the truth value of a proposition can be defined as follows:

$$\begin{aligned}&\mathit{name}: \boldsymbol{Inform}(\mathit{sender}, \mathit{receiver}, \mathit{content})\\ &\mathit{pre}: \quad \boldsymbol{makePendingComm}(\mathit{sender}, \mathit{receiver}, \mathit{content}).\mathit{precondition}\\ &\qquad\quad \mathit{and\ content.expressionType} = \text{“}\mathit{proposition}\text{”}\\ &\mathit{post}: \quad \boldsymbol{makePendingComm}(\mathit{sender}, \mathit{receiver}, \mathit{content}).\mathit{postcondition}\end{aligned}$$

The *Inform* communicative act can be performed by exchanging a message with *inform* as message type:

$$\mathit{exchMsg}(\mathit{inform}, \mathit{sender}, \mathit{receiver}, \mathit{content}) =_{\mathit{conv}} \boldsymbol{Inform}(\mathit{sender}, \mathit{receiver}, \mathit{content})$$

Notice that, according to the definition of *makePendingCommitment* provided in Sect. 3.1.1, if an agent sends the same message twice to inform a second agent about a certain state of affairs, only the first message will counts as an actual Inform. This is due to the fact that the preconditions of *makePendingCommitment* are not satisfied, since there exists a commitment with the same content among the involved agents. It is also worth observing that this does not mean that an agent cannot send multiple informs to another agent about the permanence or evolution of a certain situation, since it can change the time intervals characterizing the content of the message, represented as a temporal proposition (see Sect. 3), in such a way that preconditions of *makePendingCommitment* are satisfied.

An agent may get another agent to perform an action by performing a request, which creates an unset commitment for the receiver of the message to the performance of the action. In particular the requested action can be another communicative act. The formal definition of *Request* is the following:

$$\begin{aligned}&\mathit{name}: \boldsymbol{Request}(\mathit{sender}, \mathit{receiver}, \mathit{content})\\ &\mathit{pre}: \quad \boldsymbol{makeCommitment}(\mathit{receiver}, \mathit{sender}, \mathit{content}).\mathit{precondition}\\ &\qquad\quad \mathit{and\ content.expressionType} = \text{“}\mathit{action}\text{”}\\ &\mathit{post}: \quad \boldsymbol{makeCommitment}(\mathit{receiver}, \mathit{sender}, \mathit{content}).\mathit{postcondition}\end{aligned}$$

By sending a message with the request performative, the sender can perform the Request communicative act:

$$\mathit{exchMsg}(\mathit{request}, \mathit{sender}, \mathit{receiver}, \mathit{content}) =_{\mathit{conv}} \boldsymbol{Request}(\mathit{sender}, \mathit{receiver}, \mathit{content})$$

Unlike most content languages, temporal propositions allow us to express conditional commitments for the execution of an action *when* or *whenever* the described conditions are met (for details see [9]). Thus, in our framework *Request-When* and *Request-Whenever* coincide with *Request*.

The point of *QueryIf* is to get another agent to answer whether a state of affairs holds (the symbol $=_{\mathrm{def}}$ means that performing the action on the left-hand side is the same as performing the action on the right-hand side):

$$\begin{aligned}&\mathit{name} := \boldsymbol{QueryIf}(\mathit{sender}, \mathit{receiver}, \mathit{content}) =_{\mathit{def}} \boldsymbol{Request}(\mathit{sender},\\ &\qquad \mathit{receiver}, \boldsymbol{Inform}(\mathit{content})\ \mathit{or}\ \boldsymbol{Inform}(\mathit{not\ content}))\end{aligned}$$

This disjunctive definition of *QueryIf* is very common, but has often been criticized because it seems to suggest that the agent that is asked the question may give a positive or a negative answer independently of what it believes to be true. However, this is not the case: the answer will be produced by performing an act of informing, and by doing so the answerer will commit to the truth of what it says. Of course we cannot oblige the answerer to be sincere, but we capture the fact that every false answer will bring about the violation of a commitment.

An agent can agree to perform the requested action by executing *Agree*, which means setting to pending an unset commitment:

$$\begin{aligned}
&\mathit{name}: \boldsymbol{Agree}(\mathit{sender}, \mathit{receiver}, \mathit{content})\\
&\mathit{pre}: \quad \boldsymbol{setPending}(\mathit{sender}, \mathit{receiver}, \mathit{content}).\mathit{precondition}\\
&\qquad\quad \mathit{and\ content.expressionType} = \text{``}\mathit{action}\text{''}\\
&\mathit{post}: \quad \boldsymbol{setPending}(\mathit{sender}, \mathit{receiver}, \mathit{content}).\mathit{postcondition}
\end{aligned}$$

The creditor of an unset commitment can agree to do an action thanks to the following convention:

$$\mathit{exchMsg}(\mathit{agree}, \mathit{sender}, \mathit{receiver}, \mathit{content}) =_{\mathit{conv}} \boldsymbol{Agree}(\mathit{sender}, \mathit{receiver}, \mathit{content})$$

Otherwise, an agent that has received a request (or an act that can be defined in terms of *Request*) can refuse it, by cancelling the unset commitment. The definition of the *Refuse* communicative act and the convention for its execution are:

$$\begin{aligned}
&\mathit{name}: \boldsymbol{Refuse}(\mathit{sender}, \mathit{receiver}, \mathit{content})\\
&\mathit{pre}: \quad \boldsymbol{setCancel}(\mathit{sender}, \mathit{receiver}, \mathit{content}).\mathit{precondition}\\
&\qquad\quad \mathit{and\ content.expressionType} = \text{``}\mathit{action}\text{''}\\
&\mathit{post}: \quad \boldsymbol{setCancel}(\mathit{sender}, \mathit{receiver}, \mathit{content}).\mathit{postcondition}
\end{aligned}$$

$$\mathit{exchMsg}(\mathit{refuse}, \mathit{sender}, \mathit{receiver}, \mathit{content}) =_{\mathit{conv}} \boldsymbol{Refuse}(\mathit{sender}, \mathit{receiver}, \mathit{content})$$

As described in Section 3, a creditor of a commitment can cancel it if it has not yet reached a final state by performing the *Cancel* communicative act:

$$\begin{aligned}
&\mathit{name}: \boldsymbol{Cancel}(\mathit{sender}, \mathit{receiver}, \mathit{content})\\
&\mathit{pre}: \quad \boldsymbol{setCancel}(\mathit{sender}, \mathit{receiver}, \mathit{content}).\mathit{precondition}\\
&\qquad\quad \mathit{and\ content.expressionType} = \text{``}\mathit{action}\text{''}\\
&\mathit{post}: \quad \boldsymbol{setCancel}(\mathit{sender}, \mathit{receiver}, \mathit{content}).\mathit{postcondition}
\end{aligned}$$

$$\mathit{exchMsg}(\mathit{cancel}, \mathit{sender}, \mathit{receiver}, \mathit{content}) =_{\mathit{conv}} \boldsymbol{Cancel}(\mathit{sender}, \mathit{receiver}, \mathit{content})$$

An agent can commit itself to the performance of an action by promising it:

$$\begin{aligned}
&\mathit{name}: \boldsymbol{Promise}(\mathit{sender}, \mathit{receiver}, \mathit{content})\\
&\mathit{pre}: \quad \boldsymbol{makePendingComm}(\mathit{sender}, \mathit{receiver}, \mathit{content}).\mathit{precondition}\\
&\qquad\quad \mathit{and\ content.expressionType} = \text{``}\mathit{action}\text{''}\\
&\mathit{post}: \quad \boldsymbol{makePendingComm}(\mathit{sender}, \mathit{receiver}, \mathit{content}).\mathit{postcondition}
\end{aligned}$$

$$\begin{aligned}
&\mathit{exchMsg}(\mathit{promise}, \mathit{sender}, \mathit{receiver}, \mathit{content}) =_{\mathit{conv}}\\
&\quad \boldsymbol{Promise}(\mathit{sender}, \mathit{receiver}, \mathit{content})
\end{aligned}$$

Promise is not defined in the FIPA communicative act library. In fact, FIPA does not provide an act for communicating that an agent has an unconditional intention to perform an action. Instead, FIPA describes the *Propose* act, whose semantics states that the sender will hold the intention to execute an action if the receiver of the message accepts such intention. At present, our model does not provide any support for a commitment that should be accepted by the creditor and, for this reason, the communicative acts related with *Propose*, like *Reject-Proposal*, are not defined in our library. Furthermore, communicative act *Confirm* and *Disconfirm* cannot be defined in our model because their semantics is related to the mental state approach of FIPA-ACL. In fact, an agent perform a confirmation (or a disconfirmation) if it knows that the receiver is uncertain about a proposition.

All communicative acts previously described are defined in terms of preconditions and postconditions derived from institutional actions declared in the basic institution or in terms of other communicative acts whose definition exploits such basic actions. Thus, the agent that executes a communicative act, the sender of the message, should be authorized to perform the corresponding institutional action defined in the Basic Institution.

Finally, we consider declarations, a kind of communicative act that FIPA-ACL does not define because its semantics needs a model of institutional reality. In fact, according to Searle's Speech Act Theory [30], declarations are the particular category of communicative acts whose point is to bring about a change in the institutional reality in virtue of their successful performance. By definition the content of a declaration describes precisely the institutional changes that it brings about. Therefore we define the declaration institutional action as:

$$\begin{array}{ll} name := \boldsymbol{Declare}(sender, receiver, iaction(parameters)) & \\ pre: & iaction(parameters).precondition \\ post: & iaction(parameters).postcondition \end{array}$$

Unlike the previously defined communicative acts, which are always authorized for the sender of the message, only agents that are empowered to perform an institutional action are authorized also to declare such action:

$$\begin{array}{l} Auth(sender, \boldsymbol{Declare}(sender, receiver, iaction(parameters)), \\ \quad Auth(sender, iaction(parameters)) \end{array}$$

We introduced the use of the *Declare* institutional action (even if it has the same preconditions and postconditions of the institutional action indicated in its content) for two reasons: for uniformity with the definition of other communicative institutional actions and for similarity with the human way to communicate.

The convention that binds the exchange of a declare message to the performance of the institutional action Declare is:

$$\begin{array}{l} exchMsg(declare, sender, receiver, iaction(parameters)) =_{conv} \\ \quad \boldsymbol{Declare}(sender, receiver, iaction(parameters)) \end{array}$$

The exchange of a message of type *declare* can be considered as the universal act for the performance of institutional actions. In particular every communicative act defined in this section can be performed also by declaring it. Furthermore, given that such communicative acts are defined in terms of institutional actions on commitments, agents can obtain the same communicative effects by declaring the corresponding institutional action. This means that, at least in principle, an ACL can be defined on the basis of a single type of messages.[1]

6 English auction

In this section we will briefly describe an example (see [19] for a complete version), concerning the specification of a widely studied interaction framework: the EA. The formalization proposed exploits the conventional nature of this type of interaction, making explicit the social concepts, and rules that constitute and regulate the interaction.

In the literature there are other attempts to specify the English Auction, like for instance the one proposed by FIPA [20] and the one presented in a previous work of ours [17]. But we think that the definition of the EA as a special institution overcomes some drawbacks of those formalizations. In particular in the approach presented in [20] the commitments between the winner and the auctioneer are created only when the auction is closed. On the contrary in the current formalization and in [17] commitments are undertaken by the agents during the auction.

Another important advantage of this approach with respect to [17] is that the explicit formalization of the interaction context simplifies the content of the exchanged messages. For instance if the context is not made explicit, the auctioneer of an EA has to accept a bid of a participant, committing the auction house to give the product to that participant, on condition that no higher bids will be submitted. Otherwise the context can be made explicit for example by introducing the role *current_winner* and a norm that creates a commitment for the *current_winner* to pay the *ask_price* to the auction house, and a commitment for the auction house to give the *product* to the *current_winner* of the last round. Using this formalization, the content of the exchanged messages for bidding and for declaring the winner are simple institutional actions, as will be shown in the next section.

6.1 The english auction ontology (EAOntology)

6.1.1 Entities

The ontology of the english auction consists of some institutional entities represented as classes with the stereotype *ins-ent* in the UML class diagram described in Fig. 3. Attributes of institutional entities are depicted as class attributes and relations among entities are represented as relations among classes. Finally we introduce a stereotype *role* to represent the fact that roles may be defined relative to certain institutional entities.

The fundamental entity called *englishauction* is identified by its *id* and can assume three different states: *unset*, during the registration phase, *open*, and *closed*. An *englishauction* has a *product* that will be sold at the *ask_price*, which starts from the

[1] Carrying out a communicative act by declaration corresponds to a *performative execution* of the communicative act [32]. In human languages, however, only the communicative acts that are completely overt may have a performative execution; certain communicative acts, like for example the act of insinuating, cannot be performed by declaration, because they intrinsically contain a concealed component.

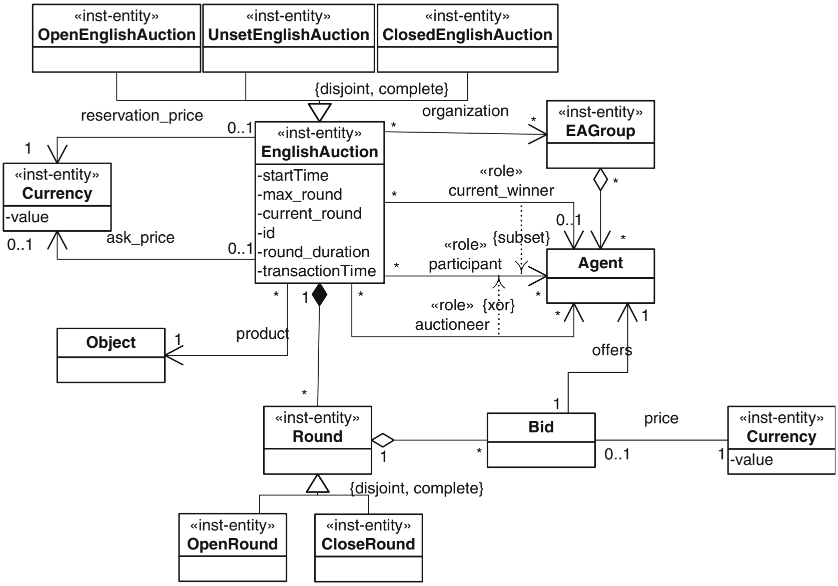

Fig. 3 Class diagram representing the english auction ontology

reservation_price and increases until there are no more bids or the maximum number of rounds (*max_round*) is reached.

An agent that takes part in an english auction can fill the role of *participant* or of *auctioneer*. In each *Round*, participants can only raise their *Bids* and the highest bidder is declared the *current_winner*. During one auction we assume that an agent cannot be both a participant and an auctioneer, while it must be a participant in order to be allowed to become a *current_winner*. Furthermore, all the agents that are related to the auction are gathered in the *EAGroup*.

Other concepts that are fundamental for every MAS, like *Agent*, are assumed to be defined in external ontologies.

6.1.2 Institutional actions

The institutional actions that operate on *EnglishAuction* entities allow agents to open and close the auction, to set the *current_winner* or a new *ask_price*, to open and close a round and to make a bid.

To describe an institutional action a slight extension of OCL is needed. In fact, using the terminology introduced in [10], object oriented specifications usually treat actions as events, because they only model state changes in the world. Instead, an action is an event brought about by an agent, and may have different effects depending on which authorized agent has performed it. For example, the act of bidding creates a new offer for the bidder and not for other agents. Therefore, we introduce a new reserved word, *actor*, that is used to refer to the agent that is performing the action. Below we formally define some of the institutional actions made available by the EA Ontology. The action for opening an auction is:

name : **openAuction**(*auct_id*)

$pre: \quad UnsetEnglishAuction.allInstances() \rightarrow exists(id = auct\_id)$
$post: \quad OpenEnglishAuction.allInstances() \rightarrow exists(id = auct\_id)$

The action for setting the *ask_price*, that can only rise, is:

$name: \mathbf{setAskPrice}(auct\_id, price)$
$pre: \quad OpenEnglishAuction.allInstances() \rightarrow exists(id = auct\_id$
$\quad and\ ask\_price.value < price)$
$post: \quad OpenEnglishAuction.allInstances() \rightarrow exists(id = auct\_id$
$\quad and\ ask\_price.value = price)$

6.2 Authorizations

Participants are authorized only to make bids by executing the *makeBid* institutional action:

$$Auth(EnglishAuction_{id}.participant, makeBid(id, price))$$

When a participant is authorized, *makeBid* is successfully executed only if the offered price is higher than the *ask_price* and if the bidder has not yet offered in the current round. Its effects are to increment the number of offers and to create a new *Bid* [19].

Auctioneers are authorized to perform all other actions defined by the EAOntology. In particular, an auctioneer is authorized to perform all institutional actions defined in Sect. 6.1.2:

$Auth(EnglishAuction_{id}.auctioneer, openAuction(id))$
$Auth(EnglishAuction_{id}.auctioneer, setAskPrice(id, price))$

6.3 Norms

Due to space limitations we will describe the main phases that characterize an auction, reporting only an example of the required norms.

To prevent system overload due to the exchange of useless messages, we assume that exists a special norm that forbids registered agents to execute those institutional actions that are ontologically impossible or that are not authorized. This norm is fired during the registration phase, whenever an agent fills a role.

When the *start_time* is elapsed and if at least two agents has been registered as participants, the following norm creates an obligation for the auctioneer to open the auction:

```
within $a$: $UnsetEnglishAuction$
on $e$: $TimeEvent(a.startTime)$
if $a.participant.sizeOf() >= 2$ then
foreach $agent$ in $a.auctioneer$
  do $makePendingComm(agent, system,$
    $(openAuction(a.id), [now, now + \delta], \exists))$
```

where δ is the time allowed to the agent to fulfill its obligation. This example shows also that our formalization of norms not only specifies correct sequences of messages, but also describes protocol timeouts. Timeouts are specified by creating an obligation for agents to perform a communicative act before a given time instant or by defining a norm that fires after a period of time and which prescribes what agents should or should not do at the next step of the interaction.

In the following phases of the interaction, the auctioneer is normally obliged to do a specific action among those it is authorized to perform, and it is forbidden from doing any of the others. Therefore, norms concerning the auctioneer have a recurrent pattern.

When the auction is declared open, a different norm creates an obligation for the auctioneer to set the reservation price and open a new round, during which the participants can bid. Unlike the auctioneer, a participant has the *permission* to make a bid but it is not obliged. Therefore, when a round is opened, its commitment not to bid has to be canceled, but no obligation to do so is created.

After the *round_duration* has elapsed, participants are prohibited from making more bids, while the auctioneer is committed to close the round, and, if there is a valid offer, to proclaim the *current_winner* and the new *ask_price*, otherwise it must close the auction.

When an agent fills the role of *current_winner*, a norm obliges it to buy the *product* on sale at the price of its last bid and forces the auctioneer to sell it. Finally, if *max_round* is not reached the auctioneer has to open a new round, otherwise it should close the auction.

7 Discussion

In this section we compare our approach to the definition of the semantics of an ACL and our model of norms with other proposals presented in this volume.

In [15] Flores et al. propose an ACL semantics based on four levels: compositional, conversational, commitment state, and joint activity. The first level, used to express the meaning of messages in term of commitments, characterized by a life cycle based on states and transitions, is very close to our proposal. However, Flores et al. do not face the problem of describing the content of messages and commitments. A further difference of their approach is that in order for a commitment to become active either the debtor or the creditor has to accept it; this is an interesting change in the life cycle of commitment that we intend to investigate in our future works. The introduction of the other levels for specifying the semantics of messages is very interesting, even if we think that these levels do not change the semantics of messages but introduce new conditions for the correctness of messages; such conditions look like to soundness conditions of interaction protocols treated in [17].

In [26] Mallya and Singh propose an algebra for combining protocols to obtain more flexible conversations. In their approach, commitments are viewed as means for engineering protocols. As we showed in Section 6, norms can describe protocols; thus it would be interesting to study in depth how different institutions can be combined, how a designer can guarantee normative coherence, or how a conversation can reach the same communicative state by following alternative paths.

In [24] Kagal and Finin present a preliminary model of conversation specification and policies using permissions and obligations. The approach to model and constrain agent interactions using normative concepts is similar to the one presented in this paper, where we describe the english auction Protocol by means of norms that restrict the set of communicative actions available for an agent at each step of the interaction. We did not distinguish between conversation specification and policies, because our norms are suitable to express both cases. Our approach is more detailed in modeling institutional reality, which is regulated by a set of norms; we also propose to distinguish between authorization (or power) and permission. Kagal and Finin's declarative approach to the formalization of norms is fit for developing a reasoning engine for artificial agents, our operational approach is particularly suitable to check if an agent is behaving in accordance with the system of norms. Finally, a very interesting topic of research faced by Kagal and Finin, that we intend to investigate in future works, is the problem of resolving conflicts between norms using for example meta-policies.

8 Conclusions

In this paper we have defined what we mean by the term "artificial institution", a description of the basic concepts that constitute agent interaction systems. Thanks to the definition of commitments as institutional entities specified by the basic institution, we give the semantics of most communicative acts defined by FIPA-ACL in terms of institutional actions on commitments. Furthermore, we have discussed the crucial role played by declarations, a communicative act that allows agents to perform every institutional action, even communicative acts. Finally we have described norms as event-driven rules that manipulate commitments, and interaction protocols as sets of norms, exemplifying our approach through a well known interaction framework, the English Auction.

We believe that our approach helps clarifying the strict relationships holding between language, institutional reality, and interaction rules in a MAS. Moreover, we believe that the adoption of an operational modelling style makes our proposal reasonably easy to implement. In fact, we plan to implement our framework as an extension of JADE in the near future.

Acknowledgements Supported by Swiss National Science Foundation project 200021-100260, "An Open Interaction Framework for Communicative Agents"

References

1. Amgoud, L., Maudet, N., & Parsons, S. (2002). An argumentation-based semantics for agent communication languages. In F. V. Harmelen (Ed.), *Proceedings of the European conference on artificial intelligence (ECAI-2002)* (pp. 38–42). Lyon, France: IOS Press.
2. Artikis, A., Kamara, L., Pitt, J., & Sergot, M. J. (2004). A Protocol for resource sharing in norm-governed Ad Hoc networks. In J. A. Leite, A. Omicini, P. Torroni & P. Yolum (Eds.), *Declarative agent languages and technologies II (DALT 2004)*, Vol. 3476 of *LNCS* (pp. 221–238). Springer, Germany.
3. Artikis, A., Pitt, J., & Sergot, M. (2002). Animated specifications of computational societies. In C. Castelfranchi & W. L. Johnson (Eds.), *Proceedings of the first international conference on autonomous agents and multiagent systems (AAMAS-02)* (pp. 1053–1062). Bologna, Italy: ACM.
4. Barbuceanu, M., Gray, T., & Mankovski, S. (1998). Coordinating with obligations. In K. P. Sycara & M. Wooldridge (Eds.), *Proceedings of the second international conference on autonomous agents* (*Agents'98*). (pp. 62–69). New York, USA: ACM.
5. Bentahar, J., Moulin, B., & Chaib-draa, B. (2004). Commitment and argument network: a new formalism for agent communication. In F. Dignum (Ed.), *Advances in agent communication, international workshop on agent communication languages, ACL 2003, Melbourne, 2003*, Vol. 2922 of *LNCS*. (pp. 146–165). Melbourne. Springer.
6. Booch, G., Rumbaugh, J., & Jacobson, I. (1999). *The unified modeling language user guide*, (First edn.). Reading, MAS, USA: Addison-Wesley.

7. Castelfranchi, C. (1995). Commitments: from individual intentions to groups and organizations. In V. Lesser (Ed.), *Proceedings of the first international conference on multi-agent systems*, (pp. 528–535). San Francisco, Cambridge, USA: MIT press.
8. Colombetti, M. (2000). A commitment-based approach to agent speech acts and conversations. In *Proceedings of the workshop on agent languages and communication policies, 4th international conference on autonomous agents (Agents 2000)* (pp. 21–29). Barcelona, Spain: Springer.
9. Colombetti, M., Fornara, N., & Verdicchio, M. (2004). A social approach to communication in multiagent systems. In J. A. Leite, A. Omicini, L. Sterling, & P. Torroni (Eds.), *Declarative agent languages and technologies I*, Vol. 2990 of *LNAI* (pp. 191–220). Germany: Springer.
10. Colombetti, M., & Verdicchio, M. (2002). An analysis of agent speech acts as institutional actions. In C. Castelfranchi, & W. L. Johnson (Eds.), *Proceedings of the first international conference on autonomous agents and multiagent systems (AAMAS-02)* (pp. 1157–1166). Bologna, Italy: ACM.
11. Dignum, F., & Royakkers, L. (1998). Collective obligation and commitment. In *In proceedings of fifth international conference on law in the information society*. Florence, Italy.
12. Esteva, M., Rodríguez-Aguilar, J. A., Sierra, C., Garcia, P., & Arcos, J. L. (2001). On the formal specification of electronic institutions. In F. Dignum, & C. Sierra (Eds.), *Agent mediated electronic commerce, the european agentlink perspective*, Vol: 1991 of *LNAI* (pp. 126–147). Berlin: Springer.
13. Esteva, M., Rodríguez-Aguilar, J. A., Sierra, C., & Vasconcelos, W. (2004). Verifying norm consistency in electronic institutions. In V. Dignum, D. Corkill, C. Jonker, & F. Dignum (Eds.), *Proceedings of the AAAI-04 workshop on agent organizations: theory and practice* (pp. 8–15). San Jose, CAL, USA. 8–15.
14. Finin, T., Labrou, Y., & Mayfield, J. (1997). KQML as an agent communication language. In J. M. Bradshaw (Ed.), *Software agents*. Chap. 14, pp. 291–316. AAAI Press Cambridge, MA: MIT Press.
15. Flores, R., Pasquier, P., & Chaib-draa, B. (2006). Conversational semantics with social commitments. *Autonomous Agents and Multi-Agent Systems* 10.1007/s10458-006-0011-1.
16. Fornara, N., & Colombetti, M. (2002). Operational specification of a commitment-based agent communication language. In C. Castelfranchi, & W. L. Johnson (Eds.), *Proceedings of the first international conference on autonomous agents and multiagent systems (AAMAS-02)* (pp. 535–542). Bologna, Italy: Springer.
17. Fornara, N. & Colombetti, M. (2003). Defining interaction protocols using a commitment-based agent communication language. In J. S. Rosenschein, T. Sandholm, M. Wooldridge, & M. Yokoo (Eds.), *Proceedings of the second international conference on autonomous agents and multiagent systems (AAMAS-03)* (pp. 520–527). Melbourne, Australia: ACM.
18. Fornara, N., & Colombetti, M. (2004). A commitment-based approach to agent communication. *Applied Artificial Intelligence an International Journal 18*(9–10), 853–866.
19. Fornara, N., Viganò, F., & Colombetti, M. (2004). Agent communication and institutional reality (extended version). Technical Report 1, Institute for Communication Technologies, Università della Svizzera Italiana.
20. Foundation for Intelligent Physical Agents (2001). FIPA english auction interaction protocol specification. http://www.fipa.org.
21. Foundation for Intelligent Physical Agents (2002). FIPA communicative act library specification. http://www.fipa.org.
22. Grossi, D., Aldewereld, H., Vazquez-Salceda, J., & Dignum, F. (2005). Ontological aspects of the implementation of norms in agent-based electronic institutions. In *Proceedings of NorMAS'05, First International Symposium on Normative Multiagent Systems. Hatfield*, UK.
23. Jones, A. & Sergot, M. J. (1996). A Formal characterisation of institutionalised power. *Journal of the IGPL 4*(3), 429–445.
24. Kagal, L., & Finin, T. (2006). Modeling conversation policies using permissions and obligations. *Autonomous Agents and Multi-Agent Systems*, in press.
25. Lopez y Lopez, F., & Luck, M. (2003). Modelling norms for autonomous agents. In Chavez, E., J. Favela, M. Mejia, & A. Oliart (Eds.), *Proceedings of fourth mexican international conference on computer science*. (pp. 238–245). CA: IEEE Computer Society.
26. Mallya, A. U., & Singh, M. P. (2006). An algebra for commitment protocols. *Autonomous Agents and Multi-Agent Systems* 10.1007/s10458-006-7232-1.
27. McBurney, P., & Parsons, S. (2003). Posit spaces: a performative model of e-commerce. In J. S. Rosenschein, T. Sandholm, M. Wooldridge, & M. Yokoo (eds.): *Proceedings of the second international conference on autonomous agents and multiagent systems (AAMAS-03)* (pp. 624–631). NY: ACM.
28. Moses, Y., & Tennenholtz, M. (1995). Artificial social systems. *Computers and AI 14*(6), 533–562.
29. Object Management Group, OMG (2003). Object constraint language specification 1.4. http://www.omg.org/.

30. Searle, J. R. (1969). *Speech acts: an essay in the philosophy of language*. Cambridge, UK: Cambridge University Press.
31. Searle, J. R. (1995). *The construction of social reality*. New York: Free Press.
32. Searle, J. R., & Vanderveken, D. (1984). *Foundations of illocutionary logic*. Cambridge, UK: Cambridge University Press.
33. Singh, M. P. (1999a). An ontology for commitments in multiagent systems: toward a unification of normative concepts. *Artificial Intelligence and Law7*, 97–113.
34. Singh, M. P. (2000). A social semantics for agent communication languages. In F. Dignum & M. Greaves (Eds.), *Issues in agent communication* (pp. 31—45), LNCS. Vol. 1916. Stockholm: Springer.
35. Vazquez, J., & Dignum, F. (2003). Modelling electronic organizations. In V. Marik, J. Muller, & M. Pechoucek. (Eds.), *Multi-agent systems and applications III*, Vol. 2691 of *LNAI* (pp. 584–593) Germany: springer.
36. Verdicchio, M., & Colombetti, M. (2003). A logical model of social commitment for agent communication. In J. S. Rosenschein, T. Sandholm, M. Wooldridge, & M. Yokoo (Eds.), *Proceedings of the second international conference on autonomous agents and multiagent systems (AAMAS-03)* pp. (528–535). Melbourne, Australia.
37. Viganò, F., Fornara, N., Colombetti, M. (2006). An event driven approach to norms in artificial institutions. In O. Boissier, J. Padget, V. Dignum, G. Lindemann, E. Matson, S. Ossowski, J. Simao Sichman, & J. Vazquez-Salceda (Eds.) Proceedings of ANIREM05 (pp. 142—154). LNAI, Vol. 3913. Utrecht, The Netherlands: Springer.
38. Viroli, M., & Ricci, A. (2005). Agent interaction semantics by timed operating instructions. In R. M. van Eijk, M. Huget, & F. Dignum (Eds.), *Agent communication, international workshop on agent communication, AC 2004, July 19, 2004, Revised Selected and Invited Papers*, Vol. 3396 of *Lecture Notes in Computer Science* (pp. 173–192). New York, NY, USA.
39. Walton, D. N., & Krabbe, C. W. (1995). *Commitment in dialogue: basic concepts of interpersonal reasoning*. Albany, NY: SUNY Press.