An Event Driven Approach to Norms in Artificial Institutions ⋆

Francesco Viganò[1], Nicoletta Fornara[1], and Marco Colombetti[1,2]

[1] Università della Svizzera italiana, via G. Buffi 13, 6900 Lugano, Switzerland
{francesco.vigano, nicoletta.fornara, marco.colombetti}@lu.unisi.ch,
[2] Politecnico di Milano, piazza Leonardo Da Vinci 32, Milano, Italy
marco.colombetti@polimi.it

Abstract. The notion of artificial institution is crucial for the specification of open and dynamic interaction frameworks where heterogeneous and autonomous agents can interact to face problems in various fields. In our view the specification of artificial institutions requires a clear standard definition of some basic concepts: the notion of ontology, authorizations, conventions, and the normative component. In this paper we propose an event driven approach to the definition of norms that is mainly based on the manipulation of commitments. We will discuss the crucial differences between the notion of authorization and permission and how the notion of permissions, obligations, and prohibitions can be expressed in our model. We will investigate the connections among the specification of different artificial institutions, in particular how an institution can enrich or further regulate the entities defined in another one. Finally we will briefly present the specification of the Dutch Auction Institution and of the Auction House Institution in order to exemplify the model presented in this paper.

1 Introduction

In the literature, the term *institution* is used with different meanings. Following organization theories, an institution can be seen as an established organization (especially of a public character) with a code of law, like for example a hospital. Drawing inspiration from economics, in multiagent systems the term *electronic institution* is commonly used to refer to a specific organization or to an abstract pattern that regulates the interaction among agents [6, 21]. In particular, institutions are viewed as means for regulating agent behavior in open and dynamic interaction systems , that is, systems where heterogeneous and autonomous agents enter and leave dynamically. In such systems norms play a crucial role because they: (i) regulate the behavior of agents, and (ii) create expectations on the behavior of other agents.

In [17], the term institution 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 ownership. Drawing inspiration from Searle's analysis of social reality, in [9] we have

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

introduced the concept of an *artificial institution* as a set of shared concepts and rules that regulate agent interactions. Artificial institutions are abstract specifications defined to obtain open interaction systems where agents can perform actions whose effects are not only limited to the state of the interaction system, but also affect human reality. For example, when an agent participates to an auction and offers an amount of money for an product on sale, if it is declared to be the winner, that product will be delivered to the address of its user, whose bank account will be charged of the negotiated amount of money. We envisage that our framework is suitable for modeling and analyzing e-business and e-government applications.

In [9] we have investigated the relation existing between agent communication and institutional reality, in particular on how agents can modify such reality. Our tenet is that agent communication changes the institutional reality existing among agents, by creating commitments between agents as in [8, 4] and also by creating new institutional states of affairs [9]. One of the most interesting aspects of our research is that we model the context where agent interactions take place and the semantics of communicative acts by means of the very same concepts. In particular, in [9] we have defined a model of institutional reality which can also be employed to describe commitments as institutional entities defined by an institution, that is, the Basic Institution.

Another important advantage of our approach is the coherence between the semantics of communicative acts and the normative component, that is, the set of norms and deontic concepts that model what agents should or should not do when their interactions are regulated by a specific institution. In fact, both communicative acts and norms are defined in terms of operations on institutional reality, in particular on social commitments, a concept widely used to define the semantics of communicative acts [19].

This paper is organized as follows. In Section 2 our view of the main components necessary for the specification of artificial institutions is presented. Among those components norms play a crucial role; our event driven model of norms based on the manipulation of commitments is discussed in Section 3. In Section 4 the connections among the specifications of different artificial institutions are investigated and in Section 5 our model is clarified through an example. Finally in Section 6 we draw some conclusions and delineate some directions for future work.

2 Artificial Institutions

To allow designers to program agents which are able to carry out institutional actions on behalf of their users in different environments, a clear and standard definition of what are the fundamental concepts of an artificial institution is needed. In our view, the specification of an artificial institution consists of the following components [9]:

- the *core ontology*, that is, the definitions of the institutional concepts introduced by the institution and of the institutional actions that operate on them;
- a set of *authorizations* specifying what agents are authorized to perform the institutional actions;
- a set of *conventions* for the concrete performance of institutional actions;
- a set of *norms* that impose obligations, prohibitions and permissions for the agents that interact within the institution.

To specify artificial institutions we adopt an operational semantics and an object-oriented style of specification, like that of the Unified Modeling Language (UML) [2] and Object Constraint Language (OCL) [16]. An advantage of this approach is the fact that it employs concepts that are close to the intuition and knowledge of practitioners. We believe the model we have developed can be easily understood by software engineers who design and implement open multiagent systems. Moreover, our model does not dictate how the system should be implemented, so that the specification of an artificial institution might be implemented on different platforms and with different languages (not necessarily object oriented), which is a fundamental requirement of open systems.

2.1 The core ontology

We assume that each institution defines an ontology describing the entities that constitute the context shared by the interacting agents and potentially affected by their communicative acts. In particular, the interaction system is modeled by a set of entities, represented through UML classes, which may have both *natural* and *institutional* attributes. While natural attributes are assumed to represent physical properties, like the size of a book, institutional attributes, like the price of a product on sale, exist only thanks to the common agreement of the agents participating to the institution. An ontology also provides a set of *institutional actions* that allow agents to change institutional attributes.

We define institutional actions by specifying:

- an action name followed by a possibly empty list of parameters: $iaction(param)$;
- a possibly empty set of (ontological) *preconditions*, which specify the values that certain institutional attributes must have;
- a nonempty set of *postconditions*, which specify the values of certain institutional attributes after a successful performance of the action.

Preconditions and effects of institutional actions are expressed through OCL formulae.

2.2 Authorizations and Conventions

Given that institutional actions modify institutional attributes, agents cannot perform such actions by exploiting causal links occurring in the natural world, like the movement of a robotic arm. Instead, we assume that all institutional actions are performed by means of a single type of instrumental actions, namely exchanging messages, thanks to the *counts-as* relation which binds the exchange of a message to the performance of an institutional action.

In order to enable counts-as relations, a set of conventions specifying what kind of message corresponds to every specific institutional action is needed. To specify what kind of message implements an institutional action we define conventions in the following form:

$$ExchMsg(msg\_type, sender, receivers, content) =_{conv} iaction(param)$$

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 must be satisfied. Firstly, an agent must be authorized to perform an institutional action. In the specification of an interaction system authorizations are expressed in term of roles, which are usually defined relative to an institutional entity. For example the role of participants and auctioneer are defined relative to the auction entity (see Section 5), and only the auctioneer can open an auction. Moreover an authorization typically holds 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. Therefore, we abstractly define the authorization to perform a specific institutional action (with given parameters) associating it to a role defined in the context of a specific institutional entity (ientity) as follows:

$Auth(ientity.role, iaction(param), conditions)$

Secondly, messages realizing institutional actions should be received by all agents that are affected by the performance of the act; for example all the participants of an auction must receive the message that opens it. Finally, ontological preconditions of institutional actions should be satisfied; for instance, an auction cannot be closed if it has not been opened yet.

If all these conditions are satisfied, the exchange of a message conventionally bound to an institutional action counts as the successful performance of such action and its institutional effects take place. In [9] we have discussed how agents can perform all types of institutional actions by means of a single message type, that is, *declare*.

3 The Normative Component

In open systems, norms have been analyzed and used from two different points of view: the design of autonomous agents [13, 7] and the design of interaction systems [20, 21]. From the second perspective norms have been exploited to indicate desirable path for the evolution of the system from an external point of view and to verify if agents are correctly behaving. In doing so, norms play an important role in a multiagent system, in that they 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 [15, 14]. But if we do not assume that norms are constraints encoded in each agent as in [15] and [14], norms are not sufficient to prevent undesirable behavior. In fact, in an open multiagent system by themselves norms are not able to banish violations, because the sincerity and benevolence of agents are not guaranteed. In this respect, our point of view is close to [7, 20, 21], where no assumptions are made about the internal structure of agents.

In our framework norms play a fundamental role, because they regulate the execution of institutional actions by an authorized agent and indicate the desired behavior imposing which actions should or should not be performed. We represent agent obligations and prohibitions as commitments that are manipulated by norms, which are treated as event-driven rules. Therefore, before presenting our conceptualization of norms, we need to introduce our model of commitment and how we describe events occurring in a multiagent system.

3.1 Commitments

In this paper we give only a short description of our model of commitment, which is the fundamental entity of what we call the *Basic Institution* [9, 4].

We regard a commitment as an institutional entity characterized by a *debtor*, a *creditor*, a *content*, and a *state*. Commitments are represented with the following notation:

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

A commitment undergoes the life cycle described in [8] by reacting either to institutional actions performed by agents or to domain-dependent events, which modify the truth value of the *temporal propositions* (for a detailed treatment see [4]), which can be *undefined*, *true*, or *false*. In [4] we have defined how the truth value of a temporal proposition is calculated. When the content of a commitment is no longer *undefined*, as a consequence of the occurrence of a domain event, the state of that commitment is automatically set to *fulfilled* if the content has become *true*, otherwise it is set to *violated*.

In our framework every agent is authorized to create a commitment by performing the *makeCommitment* institutional action, whose successful performance creates an *unset* commitment. The debtor of an *unset* commitment may refuse it by executing *setCancel*, or it may undertake the proposed commitment by executing *setPending*. We represent a refused commitment by means of the *cancelled* state, whereas an accepted commitment is depicted with the *pending* state. The creditor of *pending* or *unset* commitment can always set it to *cancelled*. Here we report the specification of another institutional action, used in the example of Section 5, *makePendingComm*, which creates a *pending* commitment and whose execution coincides with the sequential performance of *makeCommitment* and *setPending*.

$$\begin{aligned}
&name: \textbf{makePendingComm}(debtor, creditor, content)\\
&pre: \quad not\ Comm.allInstances \rightarrow exists(c|c.debtor = debtor\\
&\qquad and\ c.creditor = creditor\ and\ c.content = content)\\
&post: \quad Comm.allInstances \rightarrow exists(c|c.state = \textsf{pending}\ and\\
&\qquad c.debtor = debtor\ and\ c.creditor = creditor\ and\ c.content = content)
\end{aligned}$$

3.2 Events

As we will see, norms are event-driven rules that, when are fired by events happening in the system, modify commitments affecting the agents having a certain role. Inspired by UML notation for signals[1], here we propose to model type of events as stereotyped classes [2] 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] UML models four kinds of events: *signals, calls, passing of time* and *change in state*.

– a *TimeEvent*, that occurs when the system reaches a certain instant of time;
– a *ChangeEvent*, that happens when an institutional entity changes in some way. This kind of event type can be specialized further:
  - an *InstitutionalPropertyChange*, registered when an attribute has changed its value;
  - an *InstitutionalRelationChange*, occurring when a new relation is *created* or an existing one between the institutional entity and another one is *dropped*;
  - an *InstitutionalStateChange*, occurring when an entity modifies its type in a given taxonomy (e.g., when an auction from open becomes closed);
– an *ActionEvent*, that happens when an agent perform an action (an interesting type of this kind of events is *ExchMsg* (see Section 2.2), which represents the act of sending a message).

The definition of event types allows us to describe *event templates*, that is, event types with some restrictions on certain attributes that describe a set of possible event occurrences. Event templates are used in the *on* section of a norm to specify what kind of domain dependent events makes a norm fire.

In our experience, the specification of event templates in the definition of norms can be exploited to obtain an efficient implementation of our framework. In fact, norms should observe events occurring in the system avoiding time consuming operation to detect such changes (a similar problem is also treated in [21]). For this reason, to implement our framework we propose to apply the *observer pattern* [11], which means that objects where certain kinds of events may happen are requested to notify their *observers*, that is, those objects interested in such events, whenever an event occurs. According to the observer pattern, when a norm is interested in observing a certain kind of events, it should register at the institutional entity where they may occur. In order to reduce the number of notifications a norm receives, norms register also an event template describing what kind of events they are interested in. When an event matches an event template, the institutional entity will notify the observer that has registered it by communicating the occurred event.

3.3 Norms

We regard norms as event-driven rules that create or cancel commitments affecting a set of agents that enact a specified role within the institution. From our point of view, commitments are not a specialization of norms as in [7] and norms are not themselves a special kind of commitments as in [3] and [18]. We perceive norms as rules that manipulate commitments of the agents engaged in an interaction. In fact, norms are associated to roles rather than to individual agents, and strictly speaking they cannot be fulfilled or violated: what can be fulfilled or violated is not a norm, but rather a commitment created by the application of a norm.

At an abstract level, a norm is part of the definition of an artificial institution; its instances then regulate and are bounded by the organization that reifies the institution. When an agent fills a role in an institution, we assume that it accepts that norms create commitments binding the agent to a pseudo-agent representing the institution, which we call an *institutional agent*. Such agent allows us to keep trace of commitments created by a certain instance of institution, which also means that commitments created by norms of an institution can be canceled only by norms defined by the same institution; this is because only the creditor of a pending commitment can set it to cancelled [9]. Furthermore, if two commitments are in conflict, an agent can decide which is more important with respect to its own policy (see [13]) by reasoning about which institutional agents have created such commitments.

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 [20], 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. If certain contextual conditions, expressed through an OCL formula, are met, the activated norm is applied to a collection of *liable agents*, which are described by an OCL 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 executes a sequence of institutional actions which create or cancel commitments of the agent toward the institutional agent. 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** $liable\ agent\ selection\ expression$
  do $\{commitmentActionDescription(agent, inst\_agent, parameters)\}^+$

A crucial property of our approach is the possibility to verify at runtime if agents are compliant with a given system of norms by identifying whether they have fulfilled all commitments created by norms. Furthermore, by creating a new commitment whenever a norm is applied, we can compute how many times a norm has been violated or fulfilled by counting how many commitments instantiated by that norm are violated or fulfilled. This is important because we consider that a normative system should allow one to detect not only the presence of violations, but also how often they occur.

Using our formalization of norms, institutions can regulate in an uniform way both the communication protocol and protocol-independent normative aspects, like for instance the fulfillment of agreements made during the interaction. Norms can be used to specify protocols, because they can dictate that in certain circumstances an agent ought to send a given type of message, or react to a message in a specific way. At the same time, norms can forbid the execution of institutional actions, in particular communicative acts, even if they are authorized. Furthermore, in correspondence of events that conclude the interaction process, norms can instantiate commitments to noncommunicative actions, like the payment of the purchased goods at the price negotiated during the interaction (see section 5 and [9, 22]).

3.4 Obligations, Permissions, Prohibitions and Authorizations

In our framework commitments are used to represent all deontic relationships between agents, including as a special case the deontic relationships undertaken by the debtor through communicative acts [9]. In particular, commitments toward institutional agents are used to represent *obligations* and *prohibitions*. In general, we perceive obligations and prohibitions as commitments undertaken by an agent enacting a role within an institution, toward the institution itself; 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. Furthermore, we interpret the absence of positive or negative commitments to the execution of an action of a given type as *permissions*.

Usually in the agent literature authorization is not distinguished from permission or the former encompasses the latter [5]. For example, in [20] and [6] the authors specify agent interaction through finite state machines, which represent acts which are both authorized and permitted. In fact, *governor agents* do not allow agents to perform communicative acts that are not acceptable according with the current protocol. Similarly, in [13] and [21] norms are used to specify authorizations, which are not distinguished from permissions.

Coherently with the concept of institutionalized power of [12, 1], we distinguish between the notions of authorization and permission. The main difference between authorization and permission resides in the effects of the action. Whereas the former represents a necessary condition for the execution of institutional action, permission represents the need to regulate the performance of authorized actions, but it cannot prevent the effects deriving from the performance of a forbidden act. According to [1] an unauthorized act is performed but invalid, while in our approach is not even performed. Instead, what is successfully performed is the act of exchanging a message, which does not count as the performance of the corresponding institutional action.

4 Connections among different Artificial Institutions

We envision that when a designer starts to specify a new artificial institution, there is at his or her disposal a library of previously defined institutions that can be used to create new ones. To obtain a modular and incremental specification of new institutions, we are investigating what relations hold among artificial institutions.

When a new institution is defined by the composition of existing institutions, it can only specify new properties which refer to them but it cannot alter previous specifications. This is because institutional facts exist only thanks to common agreement, and agents involved by the former institution might not participate in the second, and thus might not access it. To preserve agents' common agreement, the new institution must regulate further aspects not regarded by others institutions and cannot be employed as an alternative or substitute of them. Drawing inspiration from UML, we have named this kind of relationship *usage relation*, which means that a *client* institution introduces several features which refer to properties defined by a *supplier* institution. In particular, a client institution may:

- define new institutional attributes relative to entities defined in a supplier institution;

– create norms concerning the performance of institutional actions described by a supplier institution; however, to keep the supplier institutions unaffected, the set of agents liable to such norms must consist of the agents enacting a role in the client institution.

The possibility of creating a new institution from preexisting institutions brings to the foreground our distinction between authorizations and permissions (i.e., the absence of prohibitions). In fact, the supplier institution may authorize the agents enacting a given role to perform a type of institutional action, whose execution is then constrained by local norms in the client institution. In this case, the agent has the necessary institutional power, but it is not permitted to perform that action by the second institution. If it executes such action, its effects take place, but the agent violates a commitment created by a norm.

5 An example: the Dutch Auction and the Auction House

In this section we will present two specification of *artificial institutions* to exemplify how norms are specified in our framework. This allows us to show how different institutions interact and to clarify the distinction between authorizations and permissions. To this purpose, we shall present our formalizations of the Dutch Auction and of the Auction House holding auctions regulated by the former institution. Due to space limitations, we focus our attention only on those aspects involved in fixing the price of the product on sale.

5.1 The Dutch Auction Institution

During a Dutch Auction an auctioneer tries to fix the price of a product. An agent taking part in a Dutch Auction can fill the role of *participant*, *auctioneer* or *transaction agent*, the agent that attends to the exchange of money and goods when a price has been accepted. During an auction we assume that a participant cannot be the auctioneer or the transaction agent, while an auctioneer might also fill the role of transaction agent.

After a period of time reserved to the registration of participants, the interaction starts when the *start time* has elapsed and the auctioneer has declared the auction open. Then, the price of the product on sale is initialized, usually higher than the expected final result. When a new price is declared, during the validity of such price, the auctioneer declares as the *winner* the first participant that accepts the *current price*, and then closes the auction. Otherwise, after the time of validity has elapsed, the auctioneer should declare a new *current price*, lower than the previous one, or close the auction.

The ontology of the Dutch Auction is described by the class diagram reported in Figure 1, where institutional entities are assigned to packages representing what institution defines them. Furthermore, concepts like *Agent*, *Product*, together with the *owner* role are imported from external ontologies. It is important to observe that the *current price* is defined relative to the Product, which means that the Dutch Auction ontology enriches the definition of such an entity by adding a new institutional attribute recognized by the agents involved in the current interaction.

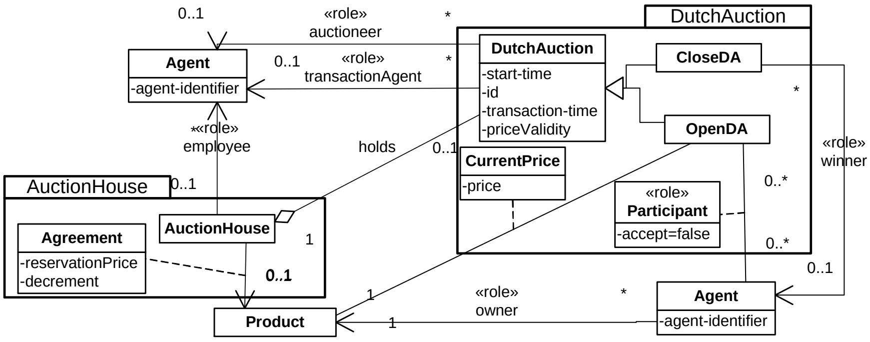

Fig. 1. The Dutch Auction and Auction House Ontologies.

The Dutch Auction ontology also defines a set of institutional actions that allow agents to operate on such institutional entities. In particular, an agent may change the current price of a Dutch Auction by performing the *setCurPrice* institutional action. We assume that an agent may change the current price of a Dutch Auction only if the auction is open and the previous price is higher than the new one, which becomes the current price of the auction.

$name$: **setCurPrice**(a_id, p)
$pre: \quad OpenDA.allInstances \rightarrow exists(id = a\_id \ and \ currentPrice.price > p)$
$post: \quad OpenDA.allInstances \rightarrow exists(id = a\_id \ and \ currentPrice.price = p)$

The Dutch Auction defines a set of authorizations for the performance of institutional actions. Some of these authorizations are conditional: for example an auctioneer is authorized to open an auction only if its *start time* has elapsed and if there are at least two agents registered as participants. Here we report only the authorization that allows the auctioneer of a given auction to perform the *setCurPrice* institutional action.

$$\textsf{Auth}(DutchAuction.allInstances \rightarrow select(id = auction\_id).auctioneer,$$
$$setCurPrice(auction\_id, price), true)$$

The behavior of agents that have joined an interaction system regulated by the Dutch Auction Institution is constrained by a normative system, which prescribes what agents should or should not do in correspondence to relevant institutional events. Likewise [9], we have defined a set of norms that regulate both the communicative acts performed by agents and the final exchange of good and money between the *transaction agent* and the *winner* of the auction (see [22]). The main advantage of our formalization with respect to the one specified by FIPA [10] is that, due to the explicit representation of norms as rules that modify agent commitments, it is possible to model in an uniform way the interaction protocols and the other rules that regulate the interaction framework. Furthermore, when an interaction terminates successfully, agents are explicitly committed by suitable norms to carry out the economic transaction.

5.2 The Auction House Institution

As reported in Figure 1, the Auction House is an institution constituted by a set of *employee* agents, holding several auctions and regulating the commercial relation with the *owner* of the product on sale. The Auction House defines only one role, *employee*, which is the role that an agent should hold in order to fill both the roles of auctioneer and transaction agent.

In order to obtain simpler and shorter OCL expressions, in this paper we will assume that an Auction House runs at most one Dutch Auction. When an agent decides to sell a product trough an auction, it reaches an *agreement* with the Auction House concerning the minimum price at which the product may be sold. Such an institutional fact does not require the agreement of the participants: in fact, participants are not even assumed to know about the existence of a reservation price (not to mention its actual current value). We regard reservation price as an institutional attribute associated to the product and representing a private agreement established between the Auction House and the *owner* of the product on sale.

A norm of the Auction House is related to the *agreement* stipulated between the owner of the good and the auction house and is activated when an employee becomes the auctioneer. This norm commits the auctioneer to not declare a price lower than agreed reservation price.

within h: $AuctionHouse$
on e: $InstitutionalRelationChange(h.dutchAuction, auctioneer, \textsf{created})$
if $true$ **then**
  foreach $agent$ **in** $h.employee \rightarrow select(em \mid e.involved \rightarrow contains(em))$
  do makePendingComm$(agent, DutchInstAgent(not\ setCurPrice($
    $h.dutchAuction.id, ?p\ [?p < h.agreement.reservationPrice]),$
    $< \textsf{now}, \textsf{now} + time\_of(e1 : InstitutionalStateChange($
    $h.dutchAuction, OpenDA, ClosedDA)) >, \forall))$

This norm is activated when an employee fills the role of auctioneer and constrains its behavior to not declare a current price lower than the reservation price, although any price would be legal from the point of view of the Dutch Auction. In fact, the Dutch Auction authorizes the auctioneer to set a new current price, imposing through the ontological preconditions of the *setCurPrice* that it should be lower than the previous one, but not further constrains are imposed.

This example shows clearly that authorizations and permissions may differ when they are relative to different sources. In fact, when a designer wants to force an agent to not perform an institutional action in correspondence of a certain state, he or she can: (i) define a new norm that creates a prohibition to not perform such action; (ii) remove the authorization to perform such act. When a designer specifies a new institution, he or she may arbitrarily choose one of these options to limit agent actions. Instead, when new institutions are defined by using previously defined institutions, agent behaviors can be conditioned only through norms, which prescribe prohibited and permitted actions.

6 Conclusions

In this paper we have presented a model for the specification of artificial institutions which clarify what are the basic concepts that must be specified in order to obtain an institution. We have focused on the conceptualization of norms as event-driven rules that modify agent commitments. The main advantage of our approach is that it employs concepts and a notation that are close to the intuition and knowledge of practitioners, and it is compatible with state-of-art software implementation techniques, in particular with *events programming*. Furthermore, we have discussed the fundamental role played by norms, which allow us to express obligations and prohibitions in terms of commitments. Indeed, norms can represent in a unified way both interaction protocols and other normative aspects. Finally we have shown, through an example, how an interaction system can be specified in terms of norms defined by different artificial institutions. In particular, we have discussed how a designer may define a new interaction framework by using several artificial institutions and what connections might exist between them.

Several research questions are still open, and will be tackled in our future work. We will investigate the development of methods for discovering inconsistencies among different artificial institutions. In particular, we are interested in verifying during the specification phase whether norms may create obligations to perform unauthorized actions, or under what conditions two norms may generate conflicting commitments.

References

1. A. Artikis, M. Sergot, and J. Pitt. Animated Specifications of Computational Societies. In C. Castelfranchi and W. L. Johnson, editor, *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2002)*, pages 535–542, Bologna, Italy, 2002. ACM Press.
2. G. Booch, J. Rumbaugh, and I. Jacobson. *The Unified Modeling Language User Guide*. Addison-Wesley, Reading, Massachusetts, USA, 1 edition, 1999.
3. C. Castelfranchi. Commitments: From Individual Intentions to Groups and Organizations. In V. Lesser, editor, *Proceedings of the 1st International Conference on Multi-Agent Systems*, pages 528–535, San Francisco, USA, 1995. AAAI-Press and MIT Press.
4. M. Colombetti, N. Fornara, and M. Verdicchio. A Social Approach to Communication in Multiagent Systems. In J. A. Leite, A. Omicini, L. Sterling, and P. Torroni, editors, *Declarative Agent Languages and Technologies*, volume 2990 of *LNAI*, pages 191–220. Springer, 2004.
5. F. Dignum. Autonomous Agents with Norms. *Artificial Intelligence and Law*, 7(1):69–79, 1999.
6. M. Esteva, J. A. Rodríguez-Aguilar, B. Rosell, and J. L. Arcos. AMELI: An Agent-based Middleware for Electronic Institutions. In N. R. Jennings, C. Sierra, L. Sonenberg, and M. Tambe, editors, *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2004)*, pages 236–243, New York, USA, 2004. ACM Press.
7. F. López y López and M. Luck and M. d'Inverno. Normative Agent Reasoning in Dynamic Societies. In N. R. Jennings, C. Sierra, L. Sonenberg, and M. Tambe, editors, *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2004)*, pages 535–542, New York, USA, 2004. ACM Press.

8. N. Fornara and M. Colombetti. A Commitment-Based Approach to Agent Communication. *Applied Artificial Intelligence an International Journal*, 18(9–10):853–866, 2004.
9. N. Fornara, F. Viganò, and M. Colombetti. Agent Communication and Institutional Reality. In van Eijk et al. [19], pages 1–17.
10. Foundation for Intelligent Physical Agents. FIPA Dutch Auction Interaction Protocol Specification. http://www.fipa.org, 2001.
11. E. Gamma, R. Helm, R. Johnson, and J. Vlissides. *Design Patterns*. Addison Wesley, 1995.
12. A. Jones and M. J. Sergot. A formal characterisation of institutionalised power. *Journal of the IGPL*, 4(3):429–445, 1996.
13. L. Kagal and T. Finin. Modeling Conversation Policies using Permissions and Obligations. In van Eijk et al. [19], pages 123–133.
14. M. Barbuceanu and T. Gray and S. Mankovski. Coordinating with Obligations. In K. P. Sycara and M. Wooldridge, editors, *Proceedings of the 2nd International Conference on Autonomous Agents (Agents'98)*, pages 62–69, New York, 1998. ACM Press.
15. Y. Moses and M. Tennenholtz. Artificial social systems. *Computers and AI*, 14(6):533–562, 1995.
16. Object Management Group. UML 2.0 OCL Specification. http://www.omg.org/, 2003.
17. J. R. Searle. *The construction of social reality*. Free Press, New York, 1995.
18. M. P. Singh. An ontology for commitments in multiagent systems: Toward a unification of normative concepts. *Artificial Intelligence and Law*, 7:97–113, 1999.
19. R. van Eijk, M. P. Huget, and F. Dignum, editors. *Agent Communication*, volume 3396 of *LNAI*. Springer Verlag, 2005.
20. W. W. Vasconcelos. Norm Verification and Analysis of Electronic Institutions. In *Proceedings of the Workshop on Declarative Agent Languages and Technologies (DALT), 3rd International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2004)*, 2004.
21. J. Vázquez Salceda, H. Aldewereld, and F. Dignum. Implementing Norms in Multiagent Systems. In I. G. Lindemann, J. Denzinger, I. J. Timm, and R. Unland, editors, *Multiagent System Technologies: Second German Conference (MATES 2004)*, volume 3187 of *LNAI*, pages 313–327, Berlin, Germany, 2004. Springer Verlag.
22. F. Viganò, N. Fornara, and M. Colombetti. An Operational Approach to Norms in Artificial Institutions. Technical Report 2, Institute for Communication Technologies, Università della Svizzera Italiana, 2005.