Norms of Conversation in a Framework for Agent Communication Languages

Rodrigo Agerri

School of Computer Science, Univ. of Birmingham
B15 2TT, Birmingham, UK
`r.agerri@cs.bham.ac.uk`

Abstract. In open and heterogeneous environments offered by the Internet, where agents are designed by different vendors, the development of standards for agent communication needs to keep abreast of new dynamic interaction modalities. The objective of this paper is to contribute to FIPA's standardization effort by proposing a pragmatic approach to the design of agent communication languages (ACLs) in which the meaning of messages is the combination of its semantics and pragmatics. First, we present a reformulation of FIPA's communicative acts (ACL semantics) using a grounded specification language which overcomes some of the usual problems attributed to FIPA's ACL semantics. Then the ACL pragmatics aims to account for the contextual factors that enriches the semantics, such agents' roles, turn-taking, and the satisfiability of messages' perlocutionary effects. We claim that the ACL pragmatics is best specified by means of norms related to agents' obligations, permissions and rights.

Keywords. Agent Communication Languages, Normative Pragmatics, Multi-Agent Systems.

1 Introduction

Enabling communication between heterogeneous agents a is crucial issue in the developing of open environments. Ideally, languages for agent communication should facilitate effective interaction without violating the autonomy and heterogeneity of the agents. This is particularly true in open environments, such as electronic commerce applications based on the Internet, where agents are designed by different constructors and work following their individual interests.

Most of the approaches to ACL semantics [1,2,3,4] are based on speech acts theory [5]. According to this theory, linguistic communication is just a special type of action that can be analyzed from three different points of view. An *illocution* is the central component of a communicative act and it corresponds to what the act intends to achieve. The illocution should be distinguished from the effect that the communicative action is meant to produce on the receiver (*perlocution*), as well as from how the actual communication is physically carried out

Dagstuhl Seminar Proceedings 07122
Normative Multi-agent Systems
http://drops.dagstuhl.de/opus/volltexte/2007/920

(*locution*). Agents communicate sending speech acts (also called *communicative acts* or *performatives* [1,2]).

Generally speaking, each speech act consists of a set of preconditions that need to hold for an agent to perform the speech act, a propositional content, and a set of perlocutionary effects (also called rational effects and post-conditions [1,2]) that encodes the effect that the speech act causes in the receiver. FIPA ACL [1] nowadays remains of the main efforts to standardization of ACLs. The definition of the speech acts is based on a *mentalistic* approach, that is, speech acts are defined in terms of agents' mental states, and the definitions of mental operators for beliefs, intentions, etc., are given in multimodal logics based on possible world semantics (SL). The main criticism to *mentalistic semantics* is that its specification language is defined using a multimodal logic which cannot be related to a computational model and therefore, it does not facilitate its pre-runtime *verification*. In relation to this, mental states in SL are not *public*, meaning that they are not verifiable by looking at the history of agents' behaviour [6,7,3]. Besides, some assumptions such as *sincerity* and *co-operativity* are rather problematic to maintain in open environments [3,8].

An answer to FIPA's shortcomings came by rethinking the general principles in agent communication and taking a social approach as opposed to a mentalistic one. From this point of view, performing a speech act produces a number of social consequences, for example agents acquiring a commitment by sending a particular message. Several authors put *commitment* as the core social notion for the specification of speech acts [3,9,10,8]. The result was the specification of *public* ACLs which, combined with the use of temporal logics [3], were a huge step forward towards the verification of ACLs.

To abandon the mentalistic concepts of goal and intention in favour of the notion of *commitment* means that the *illocutionary* aspect of communication is missing [11]. A typical case is the *request* speech act, whose illocutionary point consists of the sender having the goal or intending that the receiver execute a certain task on its behalf. The perlocutionary effect would state that the result of performing a *request* be the receiver executing the content of the speech act. Note that from a goal-based approach to communication, the consequences of the performance of a *request* affect the receiver which in this case has to either accept it or reject it. However, if we are primarily focused on the social aspects, these intuitions are not very easy to express. For example, a *request* in a commitment-based ACL would have as preconditions that "the sender commits that the receiver has committed to accepting a request from him" [3], which shows that agents' intuitive motivations when performing a request are rather odd. Furthermore, it is not easy to see how the social semantics would account for the fulfilment of the *perlocutionary effects* of performing a speech act. Dealing with autonomous agents, it is not possible to guarantee that the perlocutionary effects are satisfied in the ACL semantics, because its fulfilment depends upon the receiving agent.

As a matter of fact, we claim that trying to satisfy the perlocutionary effects by means of semantic specifications is the wrong strategy. In fact, we go further and say that if in order to explain the social consequences of performing a speech act, then the illocutionary aspect must be abandoned, we are going down the wrong path. However, leaving the fulfilment of the perlocutionary effects up to the receiver endanger the success of the communicative interchange. Therefore, a key question remains to be answered: How can we reach an equilibrium between (agents') autonomy and (communicative) efficiency?

A possible answer is to look at the interaction protocols proposed to specify and guide agents' conversations. Interaction protocols are generally concerned with the order in which speech acts are uttered. Thus, traditional approaches to conversation in agent communication do not consider the speech context (speaker, receiver, scenario, state of discourse, etc.) nor the content of the speech acts to propose the protocols [1,12,13,14]. Furthermore they adopt a *procedural* approach that reduces agent communication to an exchange of meaningless tokens [3].

Some authors [15,16] have argued in favour of a broader view of interaction protocols. They distinguished interaction protocols (also called conversation specifications) from conversation policies. The later restrict the interaction protocols based on contextual information (sender, receiver, roles, propositional content, etc.) and not only in virtue of the order. However, these approaches (notably [16]) do not give a formal and precise definition of the concepts they use in the protocols and policies. Moreover, they do not account for the interaction between the ACL semantics and pragmatics, which is necessary if we want to explain how the perlocutionary effects are to be achieved. In fact, they claim that the ACL pragmatics constitute an independent module from the semantics [16].

As an alternative we propose to modify the conception of *meaning* in agent communication. In particular, the view that meaning consists only on the specification of the ACL semantics and that the ACL pragmatics are simple protocols which give the order in which speech acts are to be used. Instead, we consider that the performance of an speech act in agent communication occurs always under certain conversational circumstances, in which agents play specific roles, respond to their own interests, etc, and that these issues should be taken into account by a full-fledged ACL pragmatics. In this sense the meaning of a speech act is the result of *using* it according to a set of rules of conversation. In this approach, the social perspective is included in the ACL pragmatics.

An ACL pragmatics based on *normative concepts* offers a convenient solution to the problems discussed above. Norms of conversation (protocols and policies) may restrict the use of certain speech acts, facilitate the achievement of the perlocutionary effects which are defined by the ACL semantics, provide mechanisms of turn-taking and take into account contextual factors to do so. Conversation norms are therefore dependent on the preconditions and perlocutionary effects established by the ACL semantics.

This work aims to be a contribution to FIPA's effort towards the standardization of agent communication. Therefore, we include in our framework a reformulation of some of the FIPA communicative acts in a speech acts library (SAL).

We use motivational notions in the definitions of the speech acts to preserve the illocutionary aspect of communication. A salient point of our approach is that the motivational operators (goals and intentions) will be interpreted *externally*. In other words, they will not refer to agents' internal mental states. Moreover, we include definitions for some categories absent in FIPA's specification (commissives and declaratives). Both the ACL semantics and pragmatics are built upon the same computational model. In this unified framework the pragmatic component accounts for the social effects of performing a speech act and thereby facilitating the achievement of its perlocutionary effects.

Next section introduces the motivations for an ACL normative pragmatics. Section 3 provides an overview of the communicative framework. In section 4, we introduce a specification language which is used in section 5 for the definition of a set of speech acts. After the specification of the ACL semantics, we discuss in section 6 two types of norms that structure conversation and we formalize the main deontic concepts in section 7. We apply their semantics in the specification of norms of conversation by means of automata using a declarative language. We finish the paper discussing some conclusions and further work.

2 Normative Pragmatics: Motivation

There is a at least a precedent in agent communication literature with respect to the view of *meaning* as the combination of the ACL semantics (speech acts) and pragmatics. Singh [3] argued that

> "What we usually refer to informally as *meaning* is a combination of the semantics and the pragmatics. We will treat the semantics as the part of the meaning that is relatively fixed and minimal. Pragmatics is the component of meaning that is context-sensitive and depends on both the application and the social structure within which is applied. [...] Pragmatic claims would be based on considerations such as the Gricean maxims of manner, quality and quantity."

Unfortunately, this paragraph does not refer to his completed work. Instead, it seems to be pointing out a direction of development in agent communication languages. This paper does assume that view of meaning and places it at the core of our proposal. ACL semantics does not fully determine the meaning of performing an speech act because the uttering and satisfiability of a speech act may depend on contextual aspects such as authority or trust of agents involved in the conversation. In this sense, we say that the ACL semantics is *underdetermined* and that pragmatic rules are required to fully determine the meaning of an speech act. Having an *underdetermined* semantics does not mean that the semantics is ambiguous, it only means that the semantic specification cannot take into account every possible scenario and linguistic interchange without loss of generality, and without violating agents' autonomy.

However, we will not follow Singh's suggestion that Gricean work on implicatures may be directly applicable to agent communication (see [17] for a preliminary Gricean approach to agent communication). Instead, we are more inclined towards a development of *normative pragmatics*. The introduction discussed the problems of a semantic-based approach to agent communication and how we may benefit from a more *unified* perspective where meaning is the combination of both the semantic and pragmatic levels. However, we still have not made an explicit point on the motivations in favour of a normative pragmatics.

In order to do so, let us assume that the semantic specification of an *inform* speech act states that when an agent i performs this act: (i) It believes its propositional content ϕ, (ii) it has the goal that the receiver j will eventually come to believe that ϕ holds, and (iii) the perlocution is that j comes eventually to believe that ϕ holds. The interpretation process would presumably be described as follows: When agent j receives the message, it will assume that the first two preconditions hold. As a consequence, j should believe that i believes that ϕ, if j trusts the sender's message, j will believe ϕ, which corresponds to the communicative goal i wanted to achieve. Assuming that agents shall do all this reasoning is, however, too idealistic. Moreover, it is computationally expensive to let agents do all this reasoning. While text recognition may be an interesting problem for computational linguistics, an agent communication language should allow agents to communicate with each other effectively and efficiently in open environments in order to achieve some goals.

The basic motivation to propose a normative pragmatics is based on the hypothesis that the reasoning described above could be avoided if we establish flexible norms of conversation (based on rights, obligations, permissions, etc.). The norms of conversation would take into account those factors that influence the satisfiability of perlocutionary effects (e.g., the receiver's responses) and it also may influence agents' behaviour according to specific circumstances.

In relation to the latter, the FIPA CAL specification provides another good example of why a normative pragmatics may be useful to regulate the use of the speech acts. The specification of the *agree* communicative act contains a *pragmatic note* that reads:

> "The precondition on the action being agreed to can include the perlocutionary effect of some other CA, such as an *inform* act. When the recipient of the agreement (for example, a contract manager) wants the agreed action to be performed, it should then bring about the precondition by performing the necessary CA. This mechanism can be used to ensure that the contractor defers performing the action until the manager is ready for the action to be done". [1, p.4]

There are a few other *pragmatic remarks* like this one throughout the FIPA CAL that are not part of the semantic specification itself. These *pragmatic remarks* point out to the need of regulating agents' use of speech acts, but the FIPA specification does not go further. The fact that the designer felt compelled to add such a note illustrates the valuable role that a normative pragmatics can play. First, it states that agents play a specific role in the interaction. Second, it constrains the behaviour of the agents in a specific context and even the timing of executing a particular action. Furthermore, norms of conversation for agent conversation should combine nicely upon normative multi-agent systems, where notions of violation and sanction, etc., are specified and – in the case of sanctions – enforced. Thus, if an agent violates some agents' rights by not following the pragmatics, a sanction mechanism is in place, providing that exists an effective relation to the general social structures and norms of the system.

Kagal and Finin [16] propose to use obligations and permissions to specify conversation policies. However, there are a number of important differences to what we are proposing: First, they do not provide a formalization for any of the deontic operators they use. Second, they claim that policies are independent of the ACL semantics, and that in fact policies should be specified in the general structure of the system. We claim the opposite for reasons given below. Third, the use of obligations produces policies that in our view are too restrictive for autonomous agents. Four, they do not consider how their policies relate to an general purpose ACL (such as FIPA ACL) for its use in open environments. Finally, they use an ontology language based on OWL as the policy specification language, but we believe that formal logic is a more suitable language to reason about normative multi-agent systems.

There is an enormous amount of work done on the theory and practice of normative multi-agent systems [18,19,20,21,22,23] traditionally related to specification of multi-agent systems using various types of deontic logic. Some of these approaches include a communicative module that allows only a domain-based interaction [19], while others have tried to build commitment-based ACLs within an institutional framework [24]. As far as we know, it is a contribution of this paper the specification and use of normative and organizational concepts to design an all-purpose unified ACL framework for agent communication, where the normative concepts are given a precise and formal definition. One of the basic concepts of our normative pragmatic approach is the notion of 'right'. Note that we are not trying to investigate what the nature of rights are, or how many different types of rights can be distinguished or anything of the like (as discussed by [20,22] among others). Instead, we give a formal definition of a notion of *right* which is convenient for communicative purposes. Thus, the meaning of 'right' in our system is restricted to this definition. We do not aim to elucidate in this paper the meaning of several deontic notions useful for the specification and reasoning about normative multi-agent systems, but to show how deontic notions can be used for specifying ACLs relevant to normative multi-agent systems.

Summarizing, an ACL Normative Pragmatics (NPRAG) shall address the effect that the following issues have on the sender's choice of speech act and the receiver's interpretation of a message:

1. **Context**: Conversation policies state the relation between participants' roles and any particular contextual information (politeness, etc.) specific of the scenario.
2. **Perlocutionary effects**: NPRAG specifies policies about agents' communicative behaviour for a given speech act.

3. Participants' methods of **turn-taking**, constructing sequences of messages across turns, and how conversation works in different conventional settings are mainly dealt with the constitutive rules or protocols of the theory.

Regarding our interests in contribution to FIPA's work on agent communication, we could have adapted the normative pragmatics with FIPA's communicative acts library to offer a unified framework for agent communication. However, we agree with most of the criticisms discussed in the introduction towards FIPA's mentalistic semantics. At the same time, we argued in favour of preserving the illocutionary component in the ACL semantics. With this aim in mind, we include in our ACL framework a grounded specification language $MLTL_I$ for the ACL semantics where the motivational operators for goals and intentions are interpreted from an *external* point of view. Moreover, the reformulation of FIPA's communicative acts using $MLTL_I$ result in a more simple and natural representation.

3 ACL Framework

There are a number of properties that an ACL framework should comply with if we want to develop a general purpose and efficient high-level communication language for multi-agent systems. We have already discussed several that are regularly mentioned in the literature. We echo the voices of authors such as [3,25,26,27] among others to propose a number of requirements that are desirable for ACLs to exhibit:

- **Autonomous**: Agent communication must endeavour in the development of artificial languages for autonomous agents.
- **Complete**: The semantics must include a wide range of speech act types, so that there are at least available those categories defined by Searle's taxonomy.
- **Contextual**: The context of FIPA ACL is fixed with the sender. This impedes to use the language in different contexts, which affects the heterogeneity of agents. Contextual factors such as agents' roles, propositional content of messages, etc., must be considered for the ACL to be applicable in a variety of scenarios.
- **Declarative**: The semantics should state the meaning of the messages, and not the order in which can be used. Guiding the use of ACLs should be done contextually. Thus, it would be possible to adapt the ACL by constraining the use of a subset in a specific context.
- **Formal**: ACL semantics and pragmatics must be formally defined. A clear and explicit specification would facilitate interoperability for open multi-agent systems.
- **Grounded**: The ACL presented should be grounded into a computational model. This will allow to translate the properties of the agents of the system into program properties. This also facilitates the verification of the ACL.

- **Public**: Communication must be public. ACL semantics must not depend on agents' private mental states. Social consequences of performing speech acts must be addressed by the ACL pragmatics.
- **Perlocutionary**: ACL pragmatics should aim to facilitate the achievement of the perlocutionary effects.

The unified ACL consists of a set of speech acts, the Speech Acts Library (SAL), and the ACL pragmatics consisting of norms that constrain agents' behaviour. We also define two specification languages, $MLTL_I$ and $NLTL_I$, to define the semantics of the cognitive and normative concepts used in the ACL. Besides, the two specification languages have a temporal component to take into account the evolution of the system over time. In this paper, the ACL semantics captures the illocutionary character of communication. The ACL pragmatics contextually regulates the use of the speech to facilitate the achievement of the perlocutionary effects. Thus, a unified ACL is defined as the tuple (we build on [26]):

$$\mathcal{UACL} = \langle SAL, MLTL_I, NPRAG, NLTL_I \rangle$$

Following FIPA CAL [1], messages of SAL are based on a STRIPS-like language with preconditions and effects. On the one hand, the preconditions have to be true for the agent to send a message (including the goal the sender intends to achieve by sending that message). On the other hand, the effects state the response that the sender wants to produce in the audience. This is a problematic issue because, as it has been already discussed, autonomous agents, by definition, cannot be forced to guarantee the effects. The semantics of SAL are given by a function

$$[\![-]\!]_{SAL} : \mathit{wff}(SAL) \rightarrow \mathit{wff}(MLTL_I)$$

The syntax of the communication language SAL is based on the FIPA ACL [1]. The semantics of the motivational and temporal operators is given by $MLTL_I$ in the next section. The language $MLTL_I$ is based on Linear Temporal Logic (LTL) extended with operators for beliefs, goals and intentions. We combine the cognitive notions with temporal operators á la Fagin *et al.* [28]. In doing so, we aim to ground $MLTL_I$ upon a computational model, the first stage to facilitate its verification [29].

In the interpretation for beliefs, goals and intentions proposed here, they are ascribed to agents by an external reasoner about the system. Following the Interpreted Systems approach [28] for modelling knowledge, agents in our framework do not compute their beliefs, goals and intentions, and as a consequence, the ACL defined using $MLTL_I$ as the semantic specification language does not rely on agents' internal (mental) states.

$NLTL_I$ consists of linear temporal operators combined with a deontic operator for obligations. $NLTL_I$ provides the semantics for the normative operators used in the specification of NPRAG. The conversation policies and interaction protocols of NPRAG can be specified using a logic-based declarative language.

4 $MLTL_I$

Traditionally, the role of formal logic in artificial intelligence and distributed computing is to provide clear formal tools to specify complex systems. However, the logic-based specifications have been criticized on the grounds that they do not provide real methodologies for building distributed systems. In order to cope with the increasing complexity of the capabilities required by agents, researchers have been using complex multimodal logics for their specification which are generally ignored by programmers that do not see a clear relation between the specifications in formal logic and computational systems [30,31].

Several authors [28,31] have argued that to bridge the gap between theory and practice, the multimodal logics used in the specification of multi-agent systems must be grounded in a computational model. There are two main semantic approaches to the formalization of agent systems via modal logics. The traditional model is based on the work of [32] on possible-world semantics. The possible-world approach includes the theory of intention [33] and the BDI logic of [34]. The problem with possible world semantics is that the accessibility relations used to define mental operators are not easily related to a computational model. Appropriate grounded semantics ensures that a clear correspondence can be found between states in the computing system and configurations in the logical description (see [29] for a good discussion on these issues).

The second approach, the Interpreted Systems model, offers a natural interpretation of the notion of knowledge in terms of states of agents in distributed systems [28]. We adapt the interpreted system approach to our purposes of giving a *grounded* and *public* semantics for beliefs, goals and intentions.

4.1 Syntax

The syntax of the language $MLTL_I$ (Motivational Linear Temporal Logic on Interpreted Systems) associated to the interpreted system IS consists of the usual vocabulary of interpreted systems IS and the accessibility relations for beliefs, goals and intentions. $MLTL_I$ structures are the result of the combination of IS with the accessibility relations $\mathcal{B}_i$, $\mathcal{G}_i$ and $\mathcal{I}_i$ defined for the structure M_I.

The following symbols and abbreviations will be used: = for definitions. To start to construct a formal language, a set of atomic propositions (where each proposition corresponds to a variable in the model) and the usual Boolean connectives are introduced: negation $\neg$, disjunction $\vee$, conjunction $\wedge$, conditional $\rightarrow$, and material equivalence $\leftrightarrow$. Atomic formulae will be denoted by ϕ, ϕ_0, ϕ_1, ψ ...

The operators X, F, G, U are called the temporal operators. All the temporal operators are interpreted relative to a *current global state*. There are many runs (sequences of global states) of the system starting at the current state. The temporal operators describe the ordering of events in time along a run and have the following intuitive meaning:

- $F\phi$ (reads "ϕ holds sometime in the future") is true of run if there exists a global state in the run where formula ϕ is true.

- $G\phi$ (reads "ϕ holds globally") is true of a if ϕ is true at every global state in the run.
- $X\phi$ (reads "ϕ holds in the next state") is true of a path if ϕ is true in the state reached immediately after the current state in the run.
- $\phi\ U\psi$ (reads "ϕ holds until ψ holds", is true of a run if ψ is true in some state in the run, and ϕ holds in all preceding states. In other words, ψ does eventually hold and that ϕ will hold everywhere until ψ holds.

Definition 1 **($MLTL_I$ Syntax).**
The syntax of the semantic specification language $MLTL_I$ is given by the following BNF expression (consider n agents):

$$\phi := AP|\neg\phi|\phi \wedge \psi|B_i\phi|G_i\phi|I_i\phi|X\phi|F\phi|G\phi|\phi U\psi$$

We use $True$ and $False$ as shorthands for $\phi \vee \neg\phi$ and $\neg True$ respectively. Although we have include in the syntax every temporal operator, we can define X, F and G as abbreviations:

$X\phi \equiv False\ U\ \phi$
$F\phi \equiv True\ U\ \phi$
$G\phi \equiv \neg F \neg\phi$

The next operator X is true at some state s whenever ϕ is true at some future point t and there are no other states between s and t. F holds if a formula is true at some point in the future and G is true always in the future, that is, there is not a future global state in which ϕ is not true.

We can conventionally establish several binding priorities for $MLTL_I$ connectives. The unary connectives ($\neg$, the temporal connectives G, F, X, and the mental attitudes operators B_i, G_i and I_i) bind most tightly. Next in priority are $\wedge$ and $\vee$, and finally $\rightarrow$ and U.

In this framework, "agent i believes ϕ" means that, "as far as agent beliefs are concerned, the system could be at a point in which ϕ holds". In other words, beliefs refer to the runs of the system. The notion of belief used in this paper does not require that the belief be true. Therefore, an agent holding a belief does not automatically made true the content of the belief. This property is central for open multi-agent systems, where agents have available incomplete and modifiable information.

An "agent i has the goal of bringing about ϕ" means that, "with respect to the agent's goals, the system could be at a point where ϕ holds". Goals can be seen as facts ϕ at a global state that an agent wants to bring about. "An agent i intending to bring about ϕ", means that from the point of view of the agents' intentions, there is a run in which i intends, along that run, to bring about ϕ.

To ascribe cognitive states from an external point of view we generate a structure M_I by associating an Interpreted System IS with a serial, transitive and euclidean structure M, so that beliefs, goals and intentions refer to *runs* of the multi-agent system. The fundamental notion in this approach is the one of *local state*. If we look the system at any point in time, every agent is in some unique *state*. The only assumptions we need to make about local states is that all the information that agents' possess of the system is encoded in their local state. Now, given that we are interested in having an ACL semantic specification language which can be used to describe the unique state of a multi-agent system at each point in time so we do not rely on the agents' internal states to evaluate and verify their communicative behaviour, we need not only to describe the local state of the agents but also the rest of the multi-agent system, which is called the *environment*. For example, when analyzing a system where agents send messages along some communication channel, useful to keep a record or history of the messages sent. Thus, when describing a multi-agent system as a whole (agents and environment), we use the notion of *global state*. These ideas are formalized in the following section where a semantics for $MLTL_I$ are defined.

4.2 Semantics

The key idea of interpreted systems is that agents are in some state at any point in time. This state is the agent's local state which consists of all the information about other agents and about the environment to which agents have access (we follow [28] in the definition of Interpreted System). Furthermore, we can also think of the whole system as being in some state. In this sense, the notion of *environment* refers to everything else in the system that is not an agent. Both the agent's local state and the environment's state conform the global state of a system.

Definition 2 (Global States).
A tuple $(s_e, s_1, \ldots, s_n)$ *represents a global state in a multi-agent system where* s_e *is the environment's state and* s_i *is agent i's local state, for* $i = 1, \ldots, n$.

A system evolves over time. Thus, a run is defined as a function from time to global states which gives a complete description of what happens over time in one possible execution of the system. Following this, a system consists of a set of runs. A system is always at a global state at some point.

Definition 3 (Runs).
A run r *over nonempty sets of global states* GS *is a sequence of global states in* GS *that gives a complete description of an execution. A point consists of a tuple* (r, m) *where* r *is a run and* m *is the time. If* $r(m) = (s_e, s_1, \ldots, s_n)$ *is the global state at point* (r, m)*, then we say that* $r_e(m) = s_e$ *and* $r_i(m) = s_i$*, for* $i = 1, \ldots, n$.

A system can be seen as a Kripke structure with no labelling or interpretation function to assign truth values to the atomic propositions.

Definition 4 (Interpreted System).
A system T *over a set of global states* GS *is a set of runs over* GS*. An interpreted system is a pair* (T, L) *where* T *is a system of runs over global states and* L *is a labelling function for the atomic propositions* AP *over* GS*, which*

assigns truth values to the atomic propositions at the global states. For every $\phi \in AP$ *and* $g \in GS$, $L(g)(\phi) \in \{true, false\}$. *A point is in the interpreted system* IS *if* $r \in T$. *Formally, an interpreted system* IS *is defined by the tuple* (T, GS_0, L).

We extend the interpreted system models with beliefs, goals and intentions. Beliefs are given a standard $KD45$ axiomatization relative to each agent. For goals and intentions, we assume a minimal KD axiomatization to ensure consistency.

Definition 5 **(M_I structure).**
Given a system of runs T, *a structure* M_I *is generated by associating the interpreted system* $IS = (T, L)$ *with the serial, transitive and euclidean Kripke structures* $M = (S, \mathcal{B}_i, \mathcal{G}_i, \mathcal{I}_i, L)$, *such that* $M_I = (GS, \mathcal{B}_i, \mathcal{G}_i, \mathcal{I}_i, L)$ *where:*

- GS *corresponds to the sets of global states in* IS.
- L *is a labelling function* $L : S \to 2^{AP}$ *from global states to truth values, where* AP *is a set of atomic propositions. This function assign truth values to the primitive propositions* AP *at each global state in* GS.
- $\mathcal{B}_i$ *where* $i = (1, \ldots, n)$ *is a set of agents, gives the accessibility relation on global states, which is serial, transitive and euclidean. Thus, we have that* $(l_e, l_1, \ldots, l_n)$ $\mathcal{B}_i$ $(l'_e, l'_1, \ldots, l'_n)$ *if* $l'_i \in GS_i$. *If* $g = (l_e, l_1, \ldots, l_n)$, $g' = (l'_e, l'_1, \ldots, l'_n)$, *and* l_i $\mathcal{B}_i$ l'_i, *then we say that* g *and* g' *are* $\mathcal{B}_i$*-accessible to agent* i. *The formula* $B_i\phi$ *is defined to be true at* g *exactly if* ϕ *is true at all the global states that are* $\mathcal{B}_i$*-accessible from* g.
- *The accessibility relations for goals* $\mathcal{G}_i$ *and intentions* $\mathcal{I}_i$ *are defined in the same manner.*

The relations for goals and intentions are serial, so we simply adopt their definition to say that the accessibility relations that characterized goals and intentions between two global states, $g\ \mathcal{G}_i\ g'$, and $g\mathcal{I}_i\ g'$ respectively, are serial. Given that $g = (s_e, s_1, \ldots, s_n)$ is the global state, we say that $g_e = s_e$ and $g_i = s_i$ for $i = 1, \ldots, n$; this means that g_i is the local state of agent i at a given time. Agents' beliefs, goals and intentions are defined with respect to their local states and can be induced to relate points. For convenience, we will sometimes use this simplified notation to refer to global states g.

We can now apply the definition of M_I to define truth of a formula ϕ at a global state $r(m)$ of the interpreted system IS.

Definition 6 **(Satisfaction in IS with respect to M_I).**
In this framework, to say that a formula ϕ *is true at a global state* $r(m)$ *in an interpreted system* IS *if it is true in the related* M_I. *Formally,*

$$(IS, r, m) \models \phi \textit{ if } (M_I, s \models \phi).$$

We would like to remark that the semantics of the accessibility relations presented here relates global states and not points. We choose global states to stress the intuitions behind interpreted systems IS. Moreover, it allows us to give a natural definition to the time operators.

Definition 7 **($MLTL_I$ semantics).**
The semantics of $MLTL_I$ is inductively defined as follows:

$(IS, r, m) \models \phi$ *iff* $L(r, m)(\phi) = true$
$(IS, r, m) \models \phi \wedge \psi$ *iff* $(IS, r, m) \models \phi$ *and* $(IS, r, m) \models \psi$
$(IS, r, m) \models \neg\phi$ *iff it is not the case that* $(IS, r, m) \models \phi$
$(IS, r, m) \models B_i\phi$ *iff* $\forall(r', m')$ *such that* $(r, m)\ \mathcal{B}_i\ (r', m')$*, then* $(IS, r', m') \models \phi$
$(IS, r, m) \models G_i(\phi)$ *iff for all* (r', m') *such that* $(r, m)\ \mathcal{G}_i\ (r', m')$*, then* $(IS, r', m') \models \phi$
$(IS, r, m) \models I_i(\phi)$ *iff for all* (r', m') *such that* $(r, m)\ \mathcal{I}_i\ (r'm')$*, then* $(IS, r'm') \models \phi$
$(IS, r, m) \models X\phi$ *iff* $(IS, r, m + 1) \models \phi$
$(IS, r, m) \models F\phi$ *iff for some time* $m' \geq m$ $(IS, r, m') \models \phi$
$(IS, r, m) \models G\phi$ *iff for all time* $m' \geq m$ $(IS, r, m') \models \phi$
$(IS, r, m) \models \phi U \psi$ *iff there is some time* $m' \geq m$ *such that along the run such that* $(IS, r, m') \models \psi$ *and for each* $m \leq m'' < m'$ *we have* $(IS, r, m'') \models \phi$*.*

There are various issues worth to comment on the semantics of $MLTL_I$: L is a labelling function on global states, that is, the truth of a primitive proposition ϕ at a state g depends only on the global state g, since the global state encapsulates all the system information at a particular point. However, there are situations, such as "agent i receiving agent j's message", where its truth does not depend on the whole global state, but only on the agents' local state. On the other hand, there are other statements which describe situations in which their truth depends on more than the global state. An statement such as "at some point in the run, the variable x is set to 5" (example from [28]) could be true at the global state g, and false at the same global state of g at a different time. This problem is solved by introducing the temporal operators, so we can easily express the idea that something is to be true in the system at some later time, namely, $F\phi$. The formula $\phi\ U\ \psi$ holds on a run if it is the case that ϕ holds continuously until ψ holds. Moreover, $\phi\ U\ \psi$ actually requires that ψ holds in some future state.

In the interpretation for beliefs, goals and intentions proposed here, these attitudes are ascribed to agents by an external reasoner about the system. In this approach, agents do not compute their beliefs, goals and intentions in any way, and as a consequence, the communication protocol defined using $MLTL_I$ does not rely on agents' private mental states. In the case of $G_i\phi$ and $I_i\phi$ the two points (r, m) and (r', m') are related if $(r'm')$ makes possible to achieve the intention (respectively, the goal) of agent i at the point (r, m).

Agents in multi-agent systems are seen as runs. In the next section we will show how $MLTL_I$ is used to externally ascribe beliefs, goals and intentions in the definition of a set of speech acts. By combining cognitive and temporal operators, we make statements about the evolution of the agents' propositional attitudes in the system. For example, we can say that agent i believes that ϕ will eventually hold along a run: $B_iF\phi$.

It is also important to remark that the semantics of $MLTL_I$ could have been presented in a different way, closer to the possible world semantics models [32], that is, by defining the accessibility relations over points of the system [35,36]. The choice of global states stresses the intuitions related to multi-agent systems.

There has been quite a lot of work in the Computer Science community on the theoretical aspects of temporal logic. In particular, the issues of decidability, complexity and axiomatizability have been largely studied. We present in the next section an axiomatization for $MLTL_I$ and discuss some issues on the complexity of reasoning about beliefs, goals and intentions with linear time. Then, we will put $MLTL_I$ into use by defining a complete set of Speech Acts.

4.3 Axiomatics

Multi-agent systems quite often operate without complete information about their environment, which could include other agents. Thus, it is interesting to use formalisms that allow us to talk of the system's changes over time. The axiomatics of $MLTL_I$ consists of the traditional $KD45_n$ for belief and KD_n for goals and intentions. i denotes a set of agents such that $i = 1, \ldots, n$.

PC All instances of propositional tautologies.
MP If ϕ and $\phi \rightarrow \psi$, then ψ.
NEC_b If ϕ, then $B_i\phi$.
NEC_g If ϕ, then $G_i\phi$.
NEC_i If ϕ, then $I_i\phi$.
K_b $B_i(\phi \rightarrow \psi) \rightarrow (B_i\phi \rightarrow B_i\psi)$.
D_b $B_i\phi \rightarrow \neg B_i \neg\phi$.
4_b $B_i\phi \rightarrow B_i B_i\phi$.
5_b $\neg B_i \neg\phi \rightarrow B_i \neg B_i \neg\phi$.
K_g $G_i(\phi \rightarrow \psi) \rightarrow (G_i\phi \rightarrow G_i\psi)$.
D_g $G_i\phi \rightarrow \neg G_i \neg\phi$.
K_i $I_i(\phi \rightarrow \psi) \rightarrow (I_i\phi \rightarrow I\psi)$.
D_i $I_i\phi \rightarrow \neg I_i \neg\phi$.

The following axioms are known to provide a sound and complete axiomatization for LTL [37].

PC All tautologies of propositional logic.
T1 $X(\phi \rightarrow \psi) \rightarrow (X\phi \rightarrow X\psi)$.
T2 $X(\neg\phi) \equiv \neg X\phi$.
T3 $\phi\ U\ \psi \equiv \psi \vee (\phi \wedge X(\phi\ U\ \psi))$.
RT1 From ϕ infer $X\phi$.
RT2 From $\phi' \rightarrow \neg\psi \wedge X\phi'$ infer $\phi' \rightarrow \neg(\phi\ U\ \psi)$.
MP From ϕ and $\phi \rightarrow \psi$ infer ψ.

The axiomatic system is denoted by the expression $(B_{KD45} G_{KD} I_{KD})_{LTL}$, which is abbreviated by $MLTL_I - Ax$.

Theorem 1. *The system $MLTL_I - Ax$ is a sound and complete axiomatization with respect to the class of models $MLTL_I$ that are serial, transitive and euclidean.*

Completeness can be shown following the technique used in [38], who gave a sound and complete axiomatization for a logic with linear time and an operator for knowledge. Furthermore, [39] has very recently given a complete axiomatization for deontic and epistemic operators with branching time. [34] also prove completeness for BDI with branching time. The sketch of the proof is as follows: We need to show that the logic complies with the finite-model property, hence it is decidable. In order to do that, we define two structures, a Hintikka structure for a given formula φ and the quotient structure for a given model. From here we can prove that φ is satisfiable by constructing a Hintikka structure for φ and we build a pseudomodel of $MLTL_I$ structures using its quotient structure. For details, we refer to the reader to the papers cited above since the length of this proof exceeds the purpose of this paper.

Our work is obviously related and influenced by the work done on linear temporal logics [40] and the interpreted systems literature [28] about knowledge. Most of the formal apparatus defined in this section will be inherited by the ACL pragmatic specification language $NLTL_I$. The main difference (if only) is that the we combine a deontic operator with the linear time component defined here.

5 Speech Acts Library (SAL)

We had three main motivations to define a semantic specification language like $MLTL_I$:

- First, given that $MLTL_I$ is going to define the semantics of the speech acts, this logic had to allow operators for beliefs, goals and intentions to express the *illocutionary* character of communication.
- Second, $MLTL_I$ had to be grounded in a computational model, so it was interesting to find an alternative to possible world semantics to include motivational attitudes in our language.
- Finally, the temporal logic component allows us to analyze how a system evolves over time.

In this section we use $MLTL_I$ to propose a *public* and *grounded* semantics. The ACL semantics consists of a Speech Acts Library which is defined using the semantic specification language $MLTL_I$. The main purpose of this semantics is to show how the different validity claims can be understood in terms of our specification language, and formalized using the logic developed. The illocutionary point of speech act are expressed in the Feasibility Preconditions (FPs). We also specify Rational Effects to capture the perlocutionary effects that the sender intends to produce on the receiver. However, note that to provide mechanisms that allow agents to achieve the Rational Effects is a task to be performed by the ACL pragmatics.

Unlike some other alternatives to FIPA ACL discussed in the introduction, we view our Speech Acts Library as a contribution to the standardization effort lead by the FIPA project. In this sense, the definition of a *public* and *grounded* semantics aims to tackle the FIPA CAL shortcomings discussed. Furthermore, in many cases the informal description of a speech act includes references such as “at some point in the future”, “once the given precondition is true”, etc. We will see that those aspects of the specification can be naturally expressed in a simpler way using $MLTL_I$. With this point in mind, we not only define at least one speech act or communicative act for each of the categories proposed by [5], but also a version for several of the communicative acts defined in the FIPA ACL is given (see [41] for a complete reformulation of *every* FIPA communicative act using $MLTL_I$).

Following [5], we classify speech acts into assertives, commissives, directives, declarations and expressives. The last category is not relevant for the purposes of this paper, so it will not be included (we are not considering *affective agents*). The syntax of the speech acts is based on the FIPA ACL. Table 1 presents our new definitions of four speech acts plus two more expressing commissives and declaratives not present in FIPA's specification.

$\langle i, inform(j, \phi)\rangle$ $FP : B_i(\phi) \wedge G_i(B_j(\phi))$ $RE : B_j\phi$	$\langle i, request(j, \phi)\rangle$ $FP : G_i(I_j(F\phi))$ $RE : F\phi$
$\langle i, confirm(j, \phi)\rangle$ $FP : B_i(\phi) \wedge B_i(B_jF\phi \vee B_jF\neg\phi))$ $RE : B_j\phi$	$\langle i, disconfirm(j, \phi)\rangle$ $FP : B_i\neg\phi \wedge B_i(B_j\phi)$ $RE : B_j\neg\phi$
$\langle i, agree(j, \phi)\rangle$ $\langle i, inform(j, (I_i\phi\ U\ \psi))\rangle$ $FP : I_i\phi\ U\ \psi$ $RE : B_j(I_i\phi\ U\ \psi)$	$\langle i, refuse(j, \phi)\rangle$ $\langle i, inform(j, \neg(I_i\phi\ U\ \psi))\rangle$ $FP : \neg(I_i\phi\ U\ \psi)$ $RE : B_j(\neg(I_i\phi\ U\ \psi))$
$\langle i, promise(j, \phi)\rangle$ $FP : I_iF\phi$ $RE : F\phi$	$\langle i, declare(j, \phi)\rangle$ $FP : G_i(X\phi)$ $RE : X\phi$

Table 1. A complete set of speech acts.

The two performatives at the top, *inform* and *request*, represent the assertives and directives respectively. *Agree* and *refuse* are included as possible exchanges after the reception of a *request*. *Declare* is an action of the declarative class and *promise* is a commissive. These last two are our contribution to the FIPA CAL specification. Therefore, adding *promise* and *declare* to the list of primitives acts in our library (SAL) together with *inform*,*request*, *confirm* and *disconfirm* results in the total number of speech acts of SAL to be twenty four [41], although it is by no means a closed catalogue.

We use Searle's taxonomy in the knowledge that there is little agreement on the number of speech acts and types which should be covered, or whether it is possible at all to provide a complete list of speech acts. In any case, this partial list of actions cover the usual communicative requirements imposed on agents. The eight speech acts provided in table 1 are representative enough of to compare FIPA's specification with respect to our own definitions.

5.1 Assertives

Assertives perform statements about the real world. The typical assertive act is *inform*. This type of actions do not intend to modify the behaviour of the receiver, but only to affect its mental states. In particular, to modify the set of beliefs the receiver holds about a proposition ϕ. The definition of *inform* proposed by FIPA ACL indicates that the sending agent believes that some proposition ϕ is true, intends that the receiving agent also believes that ϕ is true, and does not already believe that the receiver has any knowledge of the truth of ϕ. This is regarding the Feasibility Preconditions. The Rational Effect consists of the receiver coming to believe ϕ. In FIPA's formalization of this communicative act the Feasibility Precondition consists of a conjunction: The first conjunct states quite simply that agent i has to believe the proposition ϕ, and the second one states that the sender believes that the receiver does not have any knowledge of the truth of ϕ. This provided by the form $\neg B_i(Bif_j\phi \vee Uif_j\phi)$, which it is decomposed as $\neg B_i((B_j\phi \vee B_j\neg\phi) \vee (U_j\phi \vee U_j\neg\phi))$.

It seems that this precondition is too restrictive on the sender, particularly because in open environments agents may not have any information about other agents' knowledge. When someone asserts (*inform*) that ϕ, the sender usually believes that ϕ and has the goal of affecting the receiver's mental states so that it comes to believe ϕ. Any specific constrains restricting agents to perform an *inform*, until it is completely sure that the receiver does not know that ϕ, should be formulated as a conversation policy. Therefore, we propose a new definition of *inform* in table 2.

$\langle i, inform(j,\phi)\rangle$
$FP : B_i(\phi) \wedge G_i(B_j(\phi))$
$RE : B_j\phi$

Table 2. Inform.

The first part of the Feasibility Preconditions requires the sender to believe ϕ which means that we want the sender to be sincere. This is a good assumption by default, but if we want agents to be able to negotiate in competitive scenarios this may be unrealistic. A feasible solution is to specify another speech act such as *convince* that could be used when an agent *just* aims that other agent believes a proposition ϕ, irrespective of their beliefs. This could give way to a trend of defining communicative actions to be used in argumentation and negotiation scenarios.

What about the Rational Effects? The FIPA specification says that whether or not the receiver adopts the belief in the proposition ϕ will be a function of the receiver's trust in the sincerity and reliability of the sender. FIPA does not provide a method to facilitate the achievement of the Rational Effects. Besides, it is quite clear that the nature of this observation about the receiver's trust in the sincerity of the sender, etc., points out to a number of factors that transcend the ACL semantics. It seems that we may need to model, for a specific scenario, the information relative to trust and other relations between the agents. This is the role of pragmatics in natural language communication and in our view it it is also the role that a pragmatic theory should play in agent communication.

Inform is the classic assertive speech act, but there are many others. For example, answers are generally assertives. Thus, speech acts such as *agree* and *refuse* are also assertives as are *confirm* and *disconfirm*.

According to [1], *agree* is a general-purpose agreement which answers a previously received *request*. When an agent agrees then it is informing the receiver that it intends to comply with the *request*, but not until the given precondition is true. *Agree* is not a primitive, so it is formalized in terms of an *inform*:

$$\begin{array}{l} \langle i, agree(j, < i, act >, \phi) > \equiv \\ \langle i, inform(j, I_i Done(< i, act >, \phi)) > \\ FP : B_i \alpha \wedge \neg B_i (Bif_j \alpha \vee Uif_j \alpha) \\ RE : B_j \alpha \end{array}$$

The arguments of the agree performative consist of an action to be performed, *act*, and the conditions of the agreement ϕ. The conditions are analyzed as informing of the intention to do an action *act* under the condition ϕ. The condition itself has to hold for the sender to agree with the request and to execute *act*. This particular point is not very clear in the formalization. There seems to be a mismatch between the informal description of the act and the actual formal model. In any case, this type of construction is where $MLTL_I$ proves useful, because we can naturally write $I_i\phi \ U \ \psi$ to express that the sender intends to bring about ϕ until ψ along a run. More intuitively, if ψ is true, then $I_i\phi$ is is true as long as ψ holds. The conditions of agreement are expressed in a more natural way by using the temporal operator U (until), where ψ describes the fact that constitutes the precondition of the agreement at a global state $r(m)$. The second conjunct in the Feasibility Preconditions of *agree* presents the same form as in the *inform* act, so we will not repeat the point about the operators for uncertainty, knowledge and the over-specification of agents' behaviour in the ACL semantics. The same goes for the Rational Effects.

Following the above discussion, the formalization of *agree* given in table 3 tries to capture the intuition that agent i agrees with agent j to bring about some ϕ until some precondition ψ is true. This is equal to informing j that i has the intention that ϕ will eventually hold in a run until ψ holds. The FPs state that the sender has to intend that ϕ until ψ eventually holds along a run, and the REs establish that the receiver believes that the sender possess that intention.

The dual of *agree* is *refuse*. Refuse is a negative answer to a request.

According to [1], *refuse* denotes the action of refusing to perform a given action and explaining the reason for the refusal. The arguments of the performative consist of the refused action and a proposition which provides an explanation for the refusal. Moreover, *refuse* is an abbreviation of *disconfirm*: an act is possible for the agent to be performed (and providing an explanation). An agent considers that is not possible to perform an action when the action preconditions are

$$\begin{array}{l}
\langle i, agree(j, \phi)\rangle \equiv \\
\langle i, inform(j, (I_i\phi\ U\ \psi))\rangle \\
FP : I_i\phi\ U\ \psi \\
RE : B_j(I_i\phi\ U\ \psi)
\end{array}$$

Table 3. Agree.

not satisfied. As an example, an agent may be requested to perform an action for which it has insufficient privilege (hence the explanation: I have not got enough privileges).

The definition of *refuse* given by FIPA is as follows:

$$\begin{array}{l}
\langle i, refuse(j, < i, act >, \phi)\rangle \equiv \\
\langle i, disconfirm(j, Feasible(< i, act >))\rangle; \\
\langle i, inform(j, \phi \land \neg Done(< i, act >) \land \neg I_i Done(< i, act >))\rangle \\
FP : B_i \neg Feasible(< i, act >) \land B_i(B_j Feasible(< i, act >) \lor \\
U_j Feasible(< i, act >))B_i\alpha \land \neg B_i(Bif_j\alpha \lor Uif_j\alpha) \\
RE : B_j \neg Feasible(< i, act >) \land B_j\alpha
\end{array}$$

It is surprising that being *agree* and *refuse* the dual of each other their logical form does not show any similarities whatsoever. Moreover, the use of operators such as *Feasible* to provide reasons for refusing to do an action greatly complicates the complexity and decidability of the logic, as it is shown by the extremely complex definition of *refuse* given above.

Conversely, *refuse* is to be analyzed as the dual of *agree*. Following FIPA's recommendation, it is decomposed in terms of the *inform* primitive to communicate that the receiver of the request does not intend to bring about some ϕ (the object of the *request*) until ψ (the precondition of the agreement/refusal). Its definition is given by table 4.

$$\begin{array}{l}
\langle i, refuse(j, \phi)\rangle \equiv \\
\langle i, inform(j, \neg(I_i\phi\ U\ \psi))\rangle \\
FP : \neg(I_i\phi\ U\ \psi) \\
RE : B_j(\neg(I_i\phi\ U\ \psi))
\end{array}$$

Table 4. Refuse.

Formally, the precondition to send a *refuse* states that sender does not intend, along a run, to eventually bring about ϕ until ψ; the Rational Effects aims that the receiver believes that the sender does not intend to eventually bring about ϕ along a run (i.e., to fulfil the *request*) until ψ. Again, the use of temporal operators greatly simplifies the speech act definitions.

Table 1 provides a definition for two more speech acts: *confirm* and *disconfirm*. The above discussion with respect to *agree*, *refuse* and *inform* also applies to *confirm* and *disconfirm*.

In our view, the FIPA semantics are given by means of a multimodal logic with dynamic and cognitive operators (Uncertain, Feasible, Done, etc.) that greatly increases the complexity of the logic and of the speech act itself. In this sense, $MLTL_I$ greatly simplifies the speech acts definitions by using temporal operators that describe the states of the system.

With respect to the social semantics approaches, Singh [3] proposes that an *inform* means that objectively, "the sender commits that the content is true", and practically, "the sender commits that it has a reason to know the content". Singh's aim is to use commitments to make the ACL semantics public, but in doing so the idea that the sender has the specific goal that the receiver adopts a belief is missing. Another way of saying this is that the illocutionary aspect of the speech act which we defined as "what the speech act is *intended* to achieve" is lost. The analysis proposed by [10] follows similar lines to Singh, but the semantics of speech acts are not longer declarative, but they are given operationally.

5.2 Directives

The FIPA specification of the primitive *request* consists of a sender requesting the receiver to perform some action which can also be another speech act. The argument of the performative is the action that the receiver has to perform. It seems natural to think that one precondition would be that the receiver has the goal of achieving something for the sender. However, this basic aspect is not present in the FIPA definition.

We have already made the point about the complexity of the mentalistic formalizations so we will focus on the social-based proposals: Singh [3] defines *request* to objectively mean that "the sender commits that the receiver will commit to making it true" and practically that "the sender commits that the receiver has committed to accepting a request from him". Giving this meaning to a request means that the motivation to send a *request* is not clear anymore. The motivation that the sender intends to achieve a communicative goal by means of receiver agreeing to perform the action requested cannot be expressed without using motivational operators such as goals and intentions. In this sense, the use of pre-commitments [10] to analyze *requests* fails, in our view, to express that the sender explicitly expresses its interest of having the receiver executing a particular action. In this approach, a *request* is the execution of a public method which creates an empty slot that has to be filled in.

Note, however, that we have not defined actions in $MLTL_I$. Instead, the labelling function is over atomic propositions ϕ which describe the state of affairs of the system at a global state $r(m)$. However, this reflects a simple interpretation of goals: when a *request* is made, the goal of the sender is for the system to reach a particular state of affairs, which in our case, means that we request that some proposition ϕ is true at some global state $r(m)$ of the multi-agent system. This interpretation in terms of the proposition that the sender wants the receiver to achieve fits well with the intuitions about requests. This is also very similar to the intuitive meaning of goals in [42].

In this paper, when sending a *request*, the sender holds the goal of the receiver achieving a particular proposition ϕ, that is, of making true ϕ at some global state $r(m)$. Moreover, since we want the receiver to *really* try to achieve ϕ the preconditions also require that the receiver intends along a run that ϕ be eventually true. Finally, the rational effect to be achieved is that there is a run in which ϕ eventually holds.

$\langle i, request(j, \phi)\rangle$
$FP : G_i(I_j F\phi))$
$RE : F\phi$

Table 5. Request.

5.3 Commissives

Surprisingly, FIPA does not include any commissive speech acts. The traditional example of a commitment is *promise*. The sender expresses the commitment to perform the action expressed in the content of the commissive. Commissives commit the sender to perform the action uttered by the message. That is, by performing a *promise*, the sender states its intention to bring about some ϕ at some point in the system. In our approach agents *promise* to make eventually true some ϕ along a run. When sending a *promise* the sender must hold the intention of making ϕ true. The Rational Effects must be that ϕ is made true at some later point of a run.

$\langle i, promise(j, \phi)\rangle$
$FP : I_i F\phi$
$RE : F\phi$

Table 6. Promise.

5.4 Declaratives

Declaratives are not part of the FIPA CAL either. Declarations have immediate effects in an extra-linguistic institution. They are the original *performative* verbs [43]. Declarations are particularly useful for institutional actions [24]. For example, speech acts to start or terminate an interaction (conversation) are declaratives. In that kind of situations, it is necessary to identify which agents are *allowed* to perform a specific declaration. Usually, agents have the right or the permission to perform a communicative act depending on their role in the particular scenario. In an auction, for instance, the auctioneer has the right to declare the beginning of an auction. An agent wishing to participate should be given the permission (by the auctioneer) to do so. An agent may perform an action for which it has not the right to. Again, all these points are to be included in the pragmatic component of the ACL to be presented in the next sections. In the meantime we content ourselves with defining that when an agent *declares* that ϕ, it has the goal to make ϕ true in the next step of the run. The perlocution states that ϕ holds at the next step of the run. Note the use of the temporal operator X to express that in the immediate next step, ϕ holds along the run.

$\langle i, declare(j, \phi)\rangle$
$FP : G_i(X\phi)$
$RE : X\phi$

Table 7. Declare.

Note that the ACL semantics proposed has solved some of the problems summarized in Table 8. The crucial point is that $MLTL_I$ offers a grounded semantics to beliefs, goals and intentions.

Requirements	**ACLs**	
	FIPA CAL	SAL
Autonomous	?	✓
Complete	-	✓
Contextual	-	-
Declarative	✓	✓
Formal	✓	✓
Grounded	-	✓
Public	-	✓
Perlocutionary	-	-

Table 8. Requirements for ACL semantics.

The rest of the requirements state that the semantics provided by SAL respects the *autonomy* of agents, it defines a *complete* set of speech acts, it provides a *declarative* and *formal* meaning. The requirements left, that the ACL takes into account contextual factors and facilitates the achievement of the perlocutionary effects are not meet by the ACL semantics. It is the pragmatics of the language that account for the social consequences of performing an speech act by enriching speech acts minimal meaning according to the context, scenario, agents' roles, etc.

6 Constitutive and Regulative Norms

We have argued (see 2) many of the interaction protocol approaches developed so far provide a low-level procedural characterization of interactions, or are not expressive enough to take into account the contextual factors affecting communication. Still, interaction protocols are efficient using institutional contexts to model turn-taking strategies. Interaction protocols establish which sequence of messages is appropriate in each scenario. For example, in auctions, turn-taking might underlie the specific rules to ensure that they are created only when they make sense, e.g., a bidder should not make a bid prior to the advertisement.

In our approach, institutional interactions created by a FIPA interaction protocol such as an English Auction can be seen as the *constitutive rules* according to which communication takes place. Constitutive rules only establish the allowed moves within conversation. However, interaction protocols do not *regulate* or constrain the use of the speech acts according to their content and context. In order to do so, we need *regulative rules* that specify agents' rights, obligations and permissions for specific conversational contexts. This distinction between *constitutive* and *regulative* rules in communication is due to [5].

> "Some rules regulate antecedently existing forms of behaviour. For example, the rules of polite table behaviour regulate eating, but eating exists independently of these rules. Some rules, on the other hand, do not merely regulate an antecedently existing activity called playing chess; they, as it were, create the possibility of or define the activity.[...] The institutions of marriage, money, and promising are like the institutions of baseball and chess in that they are systems of such constitutive rules or conventions" [5, p.131]

In our approach, institutional speech acts are those whose meaning depend on the institution in which they are used. Normative interaction protocols correspond to the *constitutive* rules of conversations in terms of agents' rights, obligations and permissions. Additionally, *regulative* rules in agent communication deal with context-dependent aspects: Level of trust between agents, roles, content of messages and other particularities brought about the agents involved in the exchange. For example, a *politeness* rule can be specified that states agents' obligation to send a response to a request. In our framework, regulative rules are expressed by normative *conversation policies* that facilitate the achievement of the *perlocutionary effects*. Conversation policies can also affect the meaning of speech acts in institutions because the object of the rule can refer to an institutional fact. Note that the distinction between interaction protocols and policies is not new, although their relation to constitutive and regulative rules is not explicit in other approaches [9,15].

In the next section we present the pragmatic specification language $NLTL_I$ (Normative Linear Temporal Logic on Interpreted Systems). $NLTL_I$ is defined following the same methodology used for $MLTL_I$ but instead of containing cognitive operators it includes a deontic operator. Once the syntax, semantics and axiomatics of $NLTL_I$ are presented, we define the notions of violation, right and sanction, which are also to be used in the development of the interaction protocols and conversation policies.

7 $NLTL_I$

The normative temporal logic $NLTL_I$ follows the general structure of $MLTL_I$. The main difference is that while $MLTL_I$ was designed to express agents informational and deliberative states, $NLTL_I$ includes linear temporal logic combined with a deontic operator. $NLTL_I$ structures N_I are also defined by associating structures which contain deontic accessibility relations to an interpreted system IS. The definitions of run, global state, point, and the syntax of the temporal operators defined in section 4 remain the same.

The main difference of $NLTL_I$ with other the deontic logics defined to model normative multi-agent systems [18,22,44,45,21] is the fact that their semantics are based on possible worlds. Furthermore, some of these logics are highly complex due to the combination of deontic, dynamic and temporal operators.

However, there is a recent approach to deontic logic which offers a grounded semantics [30]. They define Deontic Interpreted Systems as consisting of a static interpreted system of global states of two different types: those that are allowed and those that are disallowed states of the computation. The interpreted system presented by Lomuscio and Sergot [30] is *static* in the sense that they do not include the notion of run which provides the temporal component in standard interpreted systems [28]. In a more recent work [39], a branching temporal component and two epistemic modalities are added into Deontic Interpreted Systems. $NLTL_I$ differs from the Deontic Interpreted Systems in various ways. First, we define $NLTL_I$ with respect to a interpreted system adapted to model agent communication. Second, the global states of the system are not required to be exclusively deontic. For example, we assume that information about the history of conversation, social structure, institutional facts, etc., could be encoded in the environment's state, whereas the obligations, rights, etc. of agents are to be kept in agents' local states. Third, we include a linear time component in our logic to capture the evolution of the system over time. Linear temporal logic makes the speech act specification simpler than if we were quantifying over runs. Before we present the syntax and semantics of $NLTL_I$ a few remarks on the kind of normative notions that we are interested in is offered in the next section.

7.1 Rights in Agent Communication

The central notion in the specification of norms of conversation is the concept of *right*. *Rights* give agents enough freedom, but also constrain agents' behaviour.

Intuitively, there is a middle ground between traditional obligations and permissions as defined in standard deontic logic [46], and the concept of *right* seems to be appropriate to capture that middle ground. Other definitions of right in the agents literature largely depend on the logic used.

Norman *et al.* [22] use dynamic logic to formalize a notion of right (which resemble traditional permissions) to model agreements. Alonso [44] claims that economic-based theories of rational choice, such as game theory, cannot provide a satisfactory explanation of co-operation and collective action. The reason is that in game theory, agents calculate individually their best choice. Communication does not help either, because agents do not trust each other, and will not respect any commitments. Games with multiple equilibria or with no equilibria at all also pose problems. In particular, it is not possible to reach a rational decision about the agreements agents should make. To solve this, either *ad hoc* solutions or *local points* are proposed. Boella and van der Torre [47] describe rights as sets of strategies of agents' roles. Their proposal is interesting because they argue that rights are exercised by roles, but in our view it is not clear how their idea of right is different from the set of choices that agents have available, or the set of permissions that can be specified for a specific role.

This paper does not intend to account for any possible ambiguity found in the concept of *right*, namely, about the fact that *right* is used to refer to many different things, such as having the right to live, the right to work, a right to feel proud, a right to make pre-emptive attacks, a right to vote, etc. In this sense, rights can be classified as liberties, privileges, claims, power, etc (see [20] for a detailed discussion on these issues). Instead, we are interested in a notion of right useful to a normative approach to agent communication. These interests are based on the assumption that there is a middle ground between obligations and permissions which allows coordination through communication between autonomous agents. This idea is in some sense close to what Castelfranchi [48] calls *strong permission*. A general idea of right we are interested in is provided by the following characterization [49, p.1]:

> "Rights dominate most modern understandings of what actions are proper and which institutions are just. Rights structure the forms of our governments, the contents of our laws, and the shape of morality as we perceive it. To accept a set of rights is to approve a distribution of freedom and authority, and so to endorse a certain view of what may, must, and must not be done."

An interesting point in the etymological meaning of 'right' comes from what is *fair* or *just*. This sense is used when we say that a society is "rightly ordered". When applied to individuals, rights entitle their holders to some *freedom*. For example, an agent can be entitled with the freedom to act in certain ways. In our approach, *rights* are not merely seen as the absence of obligations.

If an agent has the right to perform a speech act, then:

- It is permitted to perform it (under certain obligations), since it does not constitute a violation.

- The rest of the agents are not allowed to perform any action that violates a right-holder's action, otherwise, they are sanctioned.
- The normative system, the group, which is represented by a special type of agent, has the obligation to sanction any violation (we follow Torre *et al.* [45] on this particular point).

The function of norms in agent communication is to stabilize social interactions by making the behaviour of agents predictable to other agents of the system. Permissions are defined as the dual of obligation. Having the right to perform an speech act means that an agent must be given permission to do so and that performing that action does not constitute a violation. Not being obliged not to bring about ϕ ($\neg O \neg \phi$) does not mean that the agent has the right to bring about ϕ ($R\phi$).

The description of agent's rights and obligations can be stored and accessed by every agent at any time, so that the ACL pragmatics is public. An agent may not know whether another agent is sincere, but it can know which rights and obligations the other agent should abide to.

Using $NLTL_I$ allows us to model the evolution of agents' obligations and rights as system changes. The need of including some sort of temporality when modelling normative systems has also been defended by other authors [18,45].

7.2 $NLTL_I$ Syntax

We need to express obligations and rights within an organizational structure in which agents have roles assigned. Rights, Violation and Sanction are not defined as primitives. The only deontic primitive operator of our framework is obligation, denoted by O_i. Following the definition of the cognitive operators in the previous chapter, we will accommodate the interpretation of the primitive deontic operator for its use with respect to runs in an interpreted system.

Regarding, roles, we use the following notation:

- $i \; rr \; j$, means that i and j are role-related by rr.
- i is a member of group c, is expressed by c_i.
- r_i denotes that i plays the role r.

A role is a set of constraints that should be satisfied when an agent plays that role. For example, the role of auctioneer constrains the obligations, permissions and rights of the agent that plays that role. The scope of the role depends on the institutional reality in which it is defined (e.g., auction). A group is a set of agents (roles) that share a specific feature (i.e., being auctioneers). Finally, role relations constrain the relations between roles (e.g., the auctioneer-bidder relation).

The syntax of $NLTL_I$ consists of the vocabulary of the interpreted system IS extended with temporal operators and the deontic accessibility relation. $NLTL_I$ structures (N_I) are actually the result of the combination of IS with an accessibility relation $\mathcal{O}_i$ of a Kripke structure M.

Definition 8 **($NLTL_I$ Syntax).**
Given a finite set of agents $i = (1, \ldots, n)$*, a finite set of group names* CN*, a finite set* RN *of role names, a finite set* RR *of role relations, and a countable set* AP *of primitive propositions, the syntax is defined as follows:*

$$\varphi := AP|\neg\varphi|\varphi \wedge \psi|O_i\varphi|F\varphi|G\varphi|X\varphi|\varphi U\psi$$

Regarding the deontic operator $O_i\phi$, its traditional reading is something like "agent i is obliged to bring about ϕ", or maybe "agent i ought to bring about ϕ". It is also interesting the interpretation proposed in the Deontic Interpreted Systems (DIS) framework [30]; they define a modality $O_i\phi$ to express that "if agent i is functioning correctly, then ϕ holds". Following this, and considering the fact that our system has time built in, the deontic operator for obligation $O_i\phi$ defined in $NLTL_I$ means that "the system is at a point in which ϕ holds if agent i works (acts) correctly", which shares the same spirit that the interpretation used for for the cognitive concepts defined in $MLTL_I$.

As usual, $P_i\phi$ is the dual of $O_i\phi$ such that

$$P_i\phi = \neg O_i\neg\phi$$

Which we could gloss as "the system could be at a point in which $\neg\phi$ holds if agent i is not working (acting) correctly".

7.3 $NLTL_I$ Semantics

$NLTL_I$ structures are generated by grounding a deontic Kripke structure M into the interpreted system IS. .

Definition 9 **(Deontic Structure).**
A Deontic structure $M = (S, \mathcal{O}_i, \ldots, \mathcal{O}_n, L)$ *is serial if for any accessibility relation* $\mathcal{O}_i$ *we have that for all* s *there is a* t *such that* $(s, t) \in \mathcal{O}_i$*.*

From the Deontic structure M and IS we generate N_I structures for $NLTL_I$:

Definition 10 **(N_I structure).**
Given a system of runs T*, a structure* N_I *is generated by associating the interpreted system* $IS = (T, L)$ *with the serial Kripke structure* $M = (S, \mathcal{O}_i, L)$*, such that* $N_I = (GS, \mathcal{O}_i, L)$ *where:*

- GS *corresponds to the sets of global states in* IS*.*
- L *is a labelling function* $L : S \to 2^{AP}$ *from global states to truth values, where* AP *is a set of atomic propositions. This function assign truth values to the primitive propositions* AP *at each global state in* GS*.*
- $\mathcal{O}_i$ *where* $i = (1, \ldots, n)$ *is a set of agents, gives a serial accessibility relation on global states. Thus, we have that* $(l_e, l_1, \ldots, l_n)$ $\mathcal{O}_i$ $(l'_e, l'_1, \ldots, l'_n)$ *if* $l'_i \in GS_i$*. If* $g = (l_e, l_1, \ldots, l_n)$*,* $g' = (l'_e, l'_1, \ldots, l'_n)$*, and* l_i $\mathcal{O}_i$ l'_i*, then we say that* g *and* g' *are* $\mathcal{O}_i$*-accessible to agent* i*. The formula* $O_i\phi$ *is defined to be true at* g *exactly if* ϕ *is true at all the global states are* $\mathcal{O}_i$*-accessible from* g*.*

Definition 11 **($NLTL_I$ semantics).**

The semantics of $NLTL_I$ is inductively defined as follows:

$(IS, r, m) \models \phi$ *iff* $L(r, m)(\phi) = true$
$(IS, r, m) \models \phi \wedge \psi$ *iff* $(IS, r, m) \models \phi$ *and* $(IS, r, m) \models \psi$
$(IS, r, m) \models \neg\phi$ *iff it is not the case that* $(IS, r, m) \models \phi$
$(IS, r, m) \models O_i\phi$ *iff* $\forall(r', m')$ *such that* $(r, m)\ \mathcal{O}_i\ (r', m')$*, then* $(IS, r', m') \models \phi$
$(IS, r, m) \models X\phi$ *iff* $(IS, r, m + 1) \models \phi$
$(IS, r, m) \models F\phi$ *iff for some time* $m' \geq m$ $(IS, r, m') \models \phi$
$(IS, r, m) \models G\phi$ *iff for all time* $m' \geq m$ $(IS, r, m') \models \phi$
$(IS, r, m) \models \phi U \psi$ *iff there is some time* $m' \geq m$ *such that along the run such that* $(IS, r, m') \models \psi$ *and for each* $m \leq m'' < m'$ *we have* $(IS, r, m'') \models \phi$*.*

In the interpretation for obligations proposed here, this motivational attitude is ascribed to the agents by an external reasoner. Two points (r, m) and (r', m') are $\mathcal{O}_i$-related if $(r'm')$ makes possible that agent i functions correctly at the point (r, m). The notions of violation, right and sanction are defined as non primitives.

First, we extend the language of $NLTL_I$ to include the propositional constant V as an abbreviation of the formula defined below. The meaning of the expression $V(\phi)$ states that if ϕ holds at some point (r, m) then ϕ is considered to be a violation.

Definition 12 **(Violation).**

From each literal built from a variable ϕ*,* $V(\neg\phi)$ *means that* $\neg\phi$ *is a violation at some point* (r, m) *in the system for some* $ns \in NS$*, such that* NS *is a set of norms, iff*

$$O_i(\phi\ U\ \psi) \rightarrow (\neg\phi\ U\ \psi)$$

If the system is at a point in which ϕ holds if agent i acts correctly until ψ holds, then $\neg\phi$ holds until ψ holds. Agent i not working correctly means that ϕ does not hold and that constitutes a violation in our system. The notion of violation is of course inspired by the work of Anderson [50].

Some authors argued that undesirable states-of-affairs do not always follow infractions, and that not all violations are sanctioned. In any case, we understand the constant V as denoting a state in which some norm is violated.

Note that we have added a new element to our framework, namely, that of the normative system $ns \in NS$ that can be seen as either a a norm of the system or as a normative agent, depending on the situation. Furthermore, we model ns as the environment's local state g_e in $NLTL_I$. Thus, the environment's local state of the system will act as a normative system that assigns agents' rights, obligations and permissions, and that it is in charge of sanctioning agents when the violate a norm. We will see that our framework allows us to model the ns as an agent in charge making agents abide by the norms quite naturally.

We can imagine a context in which if an agent i is functioning correctly then it will send an *accept* message as a response to a *request* when some agreement preconditions hold. Conversely, if agent i does not *accept* the request it violates the conversation norm that specified the correct functioning of that agent (i.e., its obligations).

In some cases, it may be interesting to specify agents' behaviour by ruling that performing some action at some point does not constitute a violation. We use the notion of *right* to express this kind of norms. Thus, by using rights we specify agents' freedom to act in some specific way without that violating a norm. In this sense, *rights* are considered here exceptions to obligations. An agent has the right to bring about ϕ under some condition ψ if bringing about ϕ is not a violation ($\neg V(\phi)$). From an external point of view, we say that "there is a point in the system where agent i is functioning rightly if the holding of ϕ does not constitute a violation". We formalize this concept as follows:

Definition 13 (Right).
Let NS be a set of norms $(ns_1, \ldots, ns_n)$ encoded in the environment's local state g_e, and let the variables of agent Ag contain a set of violation variables $V(\phi)$ such that $\phi \in AP$. Agent i's functioning is right when ϕ holds, $R_i\phi$, for some $ns \in NS$ at some global state $r(m)$, $r(m) \in GS$ iff

$$\neg V\phi \; U\psi$$

Therefore, having the right to bring about ϕ under some precondition ψ means that until ψ holds along a run, then ϕ not being a violation also holds along that run. Rights are not only permissions. When an agent is exercising a right, its freedom is specified in relation to that right.

From a linguistic point of view, we can understand right-based rules as *defaults*; if law changes and an exception to a right is made, then from that point onwards exercising that particular right is considered a violation. The linguistic interpretation is that if by default an agent has the right to *agree* or *refuse* to a *request*, then there can be a new policy that overrules the default and states that from now on exercising the right to *refuse* to a *request* sent by some agent-manager is a violation of the agent-manager's rights.

So, what happens when an agent not functioning correctly or rightly brings about some ϕ, which constitutes a violation? We stated that in these cases, there is an agent ns, called the normative agent, that, if working correctly, will sanction the offending agent. The specific nature of the sanction varies from system to system, and within the same system, from one scenario to another. The general pattern, however, is that the sanctioned agent will have the obligation to do something as a punishment for its violation. For example, agent i wants to participate in a bidding process to buy a property on behalf of some estate agents. Say that to enter the auction, you need to pay some deposit of 1,000 in advance. If the agent (its role is bidder, $bidder \in RN$) wins the auction with an agreed price of 200,000 for the property but decides to break the agreement by withdrawing the bid then it is sanctioned by having the obligation to pay a fine (given that "it is not functioning correctly", that is, following the constitutive rules that define the protocol). The fine can be the 1,000 deposit paid to enter the auction. We can formalize this notion of sanction as follows:

Definition 14 (Sanction).
Let b denote the role of bidder such that $b \in RN$, then a agent i such that $i \in Ag$ playing the role of bidder b has the obligation to pay a fine (by bringing about ϕ) iff

$$b_i \wedge (O_i\phi\ U\ \psi) \wedge (\neg F\phi\ U\ \psi) \rightarrow O_i\omega$$

Thus, if the system is at a point in which if an agent playing the role b (bidder) is acting correctly, ϕ holds until ψ holds and ϕ does not eventually happens while ψ, then i is sanctioned with the obligation of paying some fine ω.

This notion of sanction presented here can be greatly complicated by considering more complex behaviour to detect and sanction violations. However, for our purposes the relatively minimal normative structure defined in this section is sufficient to formulate a normative pragmatics for agent communication. In any case, the normative specification of multi-agent systems is a difficult problem in its own, and it exceeds the purposes of this paper.

7.4 $NLTL_I$ Axiomatics

Studying the complexity of the specification language $NLTL_I$ is interesting because we do not want that protocols defined using $NLTL_I$ that are too computationally hard.

It is well-known that the system KD_n that characterizes Standard Deontic Logic is sound and complete. In this section we give a complete and sound axiomatization of $NLTL_I$ which consists of the axioms for obligations and the linear temporal component. The following axioms provide a sound and complete axiomatization of $NLTL_I$:

PC All tautologies of propositional logic.
T1 $X(\phi \rightarrow \psi) \rightarrow (X\phi \rightarrow X\psi)$.
T2 $X(\neg\phi) \equiv \neg X\phi$.
T3 $\phi\ U\ \psi \equiv \psi \vee (\phi \wedge X(\phi\ U\ \psi)$.
RT1 From ϕ infer $X\phi$.
RT2 From $\phi' \rightarrow \neg\psi \wedge X\phi'$ infer $\phi' \rightarrow \neg(\phi\ U\ \psi)$.
MP From ϕ and $\phi \rightarrow \psi$ infer ψ.

The axiomatics for the deontic operator is as follows. i denotes a set of agents such that $i = 1, \ldots, n$.

PC All instances of propositional tautologies.
MP If ϕ and $\phi \rightarrow \psi$, then ψ.
NEC If ϕ, then $O_i\phi$.
K $O_i(\phi \rightarrow \psi) \rightarrow (O_i\phi \rightarrow O_i\psi)$.
D $O_i\phi \rightarrow \neg O_i\neg\phi$.

Theorem 2. *The system $NLTL_I - Ax$ is a sound and complete axiomatization with respect to the class of models $NLTL_I$ that are serial.*

The proof of the axiomatics of $NLTL_I$ can follow the same technique as that of $MLTL_I$ (for a proof of linear temporal logics with an *S5* axiomatics for a knowledge operator see [38]).

We had various motivations to define this logic: First, given that $NLTL_I$ is going to define the semantics of the normative operators used in the conversation norms, a deontic component was needed. We have introduced an standard operator for obligation which was then used to define several other normative concepts. Among them, the notion of *right*. Second, the semantics of $NLTL_I$ are grounded upon interpreted systems. Finally, the temporal operators provide useful tools to analyze how agents' rights and obligations change over time. This also means that coordinating communication through norms allows us to focus on the external behaviour of agents, instead of modelling their mental reasoning to interpret messages.

Next section presents the interaction protocols and conversation policies that form the ACL normative pragmatics. The set of normative operators defined by $NLTL_I$ are used in the conversation norms defined in the following sections.

8 Conversation Norms

$NLTL_I$ as a specification language provides a formal, unambiguous, and grounded meaning for the key normative concepts to be used in the specification of norms of conversation. A normative point of view to agent communication can be summarized by the following points:

- Agent conversations often occur within an institution. In fact, there are specific speech acts such as *declare* that are pure institutional facts. When the appropriate role uses the adequate speech act within an institution, the agent has performed *an action* by sending that message. The rules defining the institution are *constitutive rules* specified by means of *interaction protocols*.
- Constitutive rules specify protocols such as English Auction, whereas *regulative rules* are concerned with more context-dependent aspects in the form of *conversation policies*. Both constitutive and regulative rules are declarative and their aim is to stabilize communication by contextually constraining agents' communicative behaviour.
- Agents play roles, and their roles influence their communicative behaviour thereby facilitating the achievement of the Rational Effects.
- Right is a normative notion that rules agents' communicative behaviour by specifying their freedom instead using pure restrictions and/or obligations. Furthermore, definitions of violation and sanction are provided.

The protocols and policies that conform the norms of conversation must be declarative so that they specify *what agents can achieve* using the rules instead of *how to achieve* a particular result. In our view, formal logic constitutes a more appropriate tool reason about multi-agent systems than procedural programming languages or ontology-based languages like OWL [16]. Besides, there are a number of verification techniques for logic-based specification languages

[40] of systems that can be put to good use for the the verification of agent communication languages.

When considering which language used for the specification of the speech acts library, we conclude that although the semantics of the cognitive and temporal operators were defined by $MLTL_I$, the syntax of messages was going to follow the FIPA specification. We gave two reasons for this decision: First, most of the criticisms have been addressed to its semantics. Second, we are interested in contributing to the standardization effort of agent communication led by FIPA, so we focused on solving some of the problems of semantics of FIPA CAL.

However, we cannot use the same strategy and use UML diagrams for specifying interaction protocols and conversation policies because they merely represent the order in which messages can be uttered. This paper claims that ACL pragmatics have been largely underdeveloped and it proposes a way of providing expressive and high-level normative pragmatics.

8.1 Representation

Leaving aside the procedural and diagram-based approaches already discussed, there is a recent trend in the specification of interaction protocols based on propositional linear temporal logic (PLTL) [51] and finite-state machines [52].

Endriss [51] proposes to specify the class of all sequences of messages that are allowed by a given protocol. He uses propositional LTL (PLTL) to specify the protocols and model-checking techniques to verify the runtime conformance of conversations to the protocol. Conversation templates are defined as sequences of dialogue moves (speech acts). Those dialogues that can be captured by protocols based on finite-state machines are legal according to a protocol if and only if they are accepted by the finite-state machines that correspond to the protocols.

Standard finite-state protocols and PLTL are not suitable to interactions involving commitments, social expectations and, in our case, rights and obligations. For example, we are interested in attributing to the (role of) auctioneer the obligation to close the auction at some point, and to give the bidder the right to bid after the auctioneer *declares* the auction open. In other words, we need to consider how the system evolves as a result of agents' performing actions (speech acts in our case). It is convenient that the execution of speech acts be ruled by some protocols and policies if we want communication to be efficient.

8.2 Normative Protocols and Policies

Thus, for the formulation of a high-level norms of conversation, we need to consider taking into account the following elements:

1. A set of atomic propositions P to describe facts. They usually consist of propositional content of messages.
2. A set of **agents** that participate in the conversation.
3. A set of **speech acts** (query, request, etc.) that convey the illocutionary and perlocutionary acts of performing a communicative action.

4. A set of **normative rules** of the form $np_i(sa(i, j, P))$ which consist of a normative predicate (right, obligation), the action (a speech act) and the content of the speech act ϕ.
5. A set of **broadcasting actions**. Broadcasting actions denoting events state that a speech act *sa* is sent, received, answered or not-answered. This aspect refers to the history of the conversation.
6. A set of **roles** taken by the agents involved in the interaction. Roles are specified as facts about individual agents $role_i$.
7. An agent performing the role of **normative system** *ns* encoded in the environment's local state of the system. *ns* has the obligation of monitoring the conversations to detect violations, apply sanctions and making sure that messages are delivered.

In $NLTL_I$, we formalized obligations, rights and permissions as entirely dependent on agents' local states. Thus, any communicative actions they take are a function of their local state. Their local states also contain information regarding their initial state in the execution and the history of messages sent and receive (i.e., its conversational record; we build on the knowledge-based interpreted system model [28] to model the history of conversation).

Definition 15 (History).

Let us consider an agent i such that $i \in Ag$, a set of broadcasting actions BE, a set of speech acts SA, a set of initial states S_{0i} for agent i, and a set of contextual actions DO_i for i. A history for agent i is a sequence where

1. *The first element is in S_{0i},*
2. *the later elements consist of nonempty sets of broadcasting actions such as $sent_i(sa(i, j, P))$, $receive_i(sa(i, j, P))$, or $do(i, \alpha)$ such that $\alpha \in DO_i$.*

The history of conversation of an agent i at some point (r, m) of the system is composed by its initial state and the sequence of steps corresponding to i's actions up to time m. We can also say that if an agent i at a point (r, m) has only sent an *agree* speech act to agent j, $sent_i(agree(i, j, P))$, then its history at point (r, m) is the result of appending the set $\{sent(i, j, agree(P))\}$. Furthermore, a broadcasting event occurs in round $m + 2$ of run r if it is contained in some agent's history of conversation in $(r, m + 2)$.

We have mentioned above that our framework models the system environment as a normative agent *ns* whose task is to decide when performing a speech act is a violation and the sanctioning it when appropriate. In order to take these decisions the environment's local state must record the events that take place in the system, namely, the speech acts performed by the agents involved in a conversation. Furthermore, it need to keep an up to date record of the evolution of agents' rights, obligations and permissions according to the actions they have performed so far, taking into account the fact that performing speech acts' cause social expectations. Note, however, that determining and reasoning about the actions that *ns* can perform is part of the social structure of the system. Therefore, the ACL specification does not account for the acquisition of knowledge or beliefs by ns nor the reasoning employed to sanction violations. Doing so is not within the purposes of this paper.

Thus, we need to consider both agents' and the environment's actions to explain how their actions cause the system to change state: $(\alpha_e, \alpha_1, \ldots, \alpha_n)$ and a transition function that maps global states to global states: $\delta(\alpha_e, \alpha_1, \ldots, \alpha_n)$. We can now define a protocol as a mapping from the set L_i of agent i's local states to nonempty sets of acts in BE_i. Furthermore, a protocol P_e for the normative agent ns is a mapping from the set of the environment's local states L_e to nonempty sets of actions in DO_e.

We include normative concepts and propositional variables in our protocol rules. Furthermore, these rules must be declarative, that is, they say what the rights and permissions of the agents are, rather than a procedure to move from to one state to another. This secures the high-level character of our ACL. Interaction protocols are defined in NPRAG using if-then rules as the constitutive rules that specify the legal interactions of conversations. If agent j receives a *request* then agent j has the right to answer either by *agreeing* or by *refusing.*

We elaborate on these points in order to give specify some of the FIPA interaction protocols.

8.3 Request

Typically, protocols are described by means of programs written in some programming language. For clarity of exposition we will use in this paper $NLTL_I$ extended with parameters for agents, roles and actions. Having extended the Interpreted Systems model to express normative notions for their use in agent communication languages, we could have employed a similar strategy and adapt a simple programming language defined within the interpreted systems model [28] to express protocols that include agents' roles, rights, obligations, speech acts and broadcasting actions. After showing in this section how our approach can be used to specify an ACL pragmatics using norms, we will offer an example of a protocol using a simple programming language.

Let us consider again the FIPA Request interaction protocol. This protocol allows one agent to request to bring about some propositional content ϕ. If the receiver of the *request* speech act is functioning *rightly*, then it will send an *agree* or a *refuse* as a response to the *request.* If the answer is an *agree*, and the agent is functioning correctly at that point, then it will communicate an *inform* if the request is satisfied, or a *failure* if the object of the request is not achieved. The specification of this protocol in NPRAG looks is composed by the following norms of conversation:

1. $principal_i \wedge secretary_j \rightarrow R_i(request(i, j, \phi))$
2. $receive_j(request(i, j, \phi)) \wedge \neg sent_j(refuse(j, i, \phi)) \rightarrow R_j(refuse(j, i, \phi))$
3. $receive_j(request(i, j, \phi)) \wedge \neg sent_j(agree(j, i, \phi)) \rightarrow R_j(agree(j, i, \phi))$
4. $sent_j(agree(j, i, \phi)) \wedge F\phi \rightarrow O_j(inform(j, i, \phi))$
5. $sent_j(agree(j, i, \phi)) \wedge \neg F\phi \rightarrow O_j(failure(j, i, \phi))$

Note that the proposition of the normative predicates for rights, obligations and permissions are taken as expressing a communicative action like "agent i agrees with agent j to bring about some ϕ".

In the Request specification there are two agents i and j that take the roles of *secretary* and *principal* respectively. As a propositional content of the speech acts, we can think of a situation in which agent *principal* has the right to request to agent *secretary* to book a number of flights.

The rules state that the *principal* has the right to send any request message to the secretary, and that the secretary can answer to these messages either by agreeing or refusing if an answer has not been produced yet. The two obligation rules state that an agent has the obligation to send an *inform* having already sent an *agree* message and not having sent yet *inform* that the request has been satisfied.

As it is, the reasoning rules presented above capture the transitions that a system functioning rightly can perform under the NPRAG Request interaction protocol. However, we need something else, that is, to instantiate some of the facts of the NPRAG specification of *request.* In particular, we need to say which messages have been sent or are still pending. As discussed above, the history of conversation is part of agents' local state, whereas the status of messages and agents' rights and obligations are encoded in the environment's local state. None of these components are part of the interaction protocol specification. Indeed, for the sake of generality, it is desirable that our protocols only provide a set of norms of conversation to facilitate agents' next move *in absence of any specific circumstances.*

8.4 Query-If

In the FIPA Query-IF interaction protocol, an agent i queries agent j whether or not a proposition ϕ is true. The receiver has the right to either *agree* or *refuse* to send and *inform* message providing an answer. In the case that agent j agrees, then it has obligation to send a notification which can be an inform stating the truth of falsehood of the proposition ϕ. If agent j sends a refuse message the protocol ends there. We only show the relevant normative rules of this protocol:

1. $journalist_i \wedge policitian_j \rightarrow R_i(queryif(i,j,\phi))$
2. $receive_j(queryif(i,j,\phi)) \wedge \neg sent_j(refuse(j,i,\phi)) \rightarrow R_j(refuse(j,i,\phi))$
3. $receive_j(queryif(i,j,\phi)) \wedge \neg sent_j(agree(j,i,\phi)) \rightarrow R_j(agree(j,i,\phi))$
4. $sent_j(agree(j,i,\phi)) \wedge F\phi \rightarrow O_j(inform(j,i,\phi))$
5. $sent_j(agree(j,i,\phi)) \wedge \neg F\phi \rightarrow O_j(failure(j,i,\phi))$

We can see that its structure is almost equivalent to the Request protocol; only the use of *queryif* instead of *request* is different. This means that our proposal is high-level enough so that it is easily adaptable to represent different interaction protocols and different contexts. Only the content of the messages and the roles of the agents may change.

The specification of the constitutive rules of conversations enable us to formulate a number of policies that contextually contrain agents' communicative behaviour within the protocol in terms of their rights, obligations and permissions.

8.5 Conversation Policies

Since conversation policies usually restrict agents' behaviour within conversations, the notation of the pragmatic regulative rules that conform NPRAG conversation policies consists of the components used in the specification of interaction protocols. Moreover, we would like to stress the importance of one of the elements and propose a new one:

- A set of **contextual actions** DO_i that depend on specific scenarios, e.g., the action of *bidding* depends on the agent being in an auction.
- A **conflict resolution action** so that in case of conflict between rules of a policy, one rule has *priority* over another one.

Constructs such as the conflict resolution actions, the contextual and broadcasting actions depend on the platform in which agents run. That is, these actions are defined by the programming language in which agents are built. For example, in Java built platforms like JADE, sending messages is simply a case of creating an ACLMessage, setting the parameters (sender, receiver, reply-to, performative, etc.) and then sending it using the send() method in the agent object.

If the normative rules in the interaction protocols specify the legal structure of the conversation, conversation policies regulate agents' behaviour according to contextual information within the protocol. Roles and background knowledge provide valuable information for agents to choose the right course of action. Unlike the specification of the interaction protocols, we consider the content of the speech acts when proposing normative rules. Furthermore, note that the policies are tightly combined with the ACL semantics defined in the previous chapter. Thus, the meaning of a speech act such as *queryif* is enriched by the rights, obligations and permissions of agents to use that particular speech act.

We can imagine a situation in which an agent *paxman* has the right to *queryif* a politician agent *pm* about the truth of the "peersmoney" scandal as long as we are not in electoral campaign.

$$paxman_i \wedge pm_j \rightarrow R_i(queryif(i, j(peersmoney))U\neg(elections)$$

Another example can be of an agent j acting on behalf of an airline company serving flights to European countries, that could have a policy that states that it should agree to every request regarding flight tickets to Europe (i.e., answering about flight times and providing the best offer for a potential buyer) and another one specifying that it has the obligation to refuse every request about flights to non European countries.

- $customer_i \wedge seller_j \wedge receive_j(request(i, j, europeanFlight)) \rightarrow O_j(agree(j, i, \phi))$.
- $receive_j(request(i, j, nonEuropeanFlight)) \rightarrow O_j(refuse(j, i, \phi))$.

This issue shows how using normative conversation policies help agents to achieve the perlocutionary effects since the perlocution of *agree*, namely, that the receiver satisfies the object of the requested action, is now specified to be an obligation of the seller. This is a crucial point to help agents to achieve the rational effects of an speech act. For example, we can specify a rule to state that if an agent makes a promise to increase the taxes on air planes fuel, then it has the obligation to do so:

$$\begin{gathered} G(send_i(promise(i, public, taxairplanesFuel)) \rightarrow \\ O_i(increaseTaxes(airplanesFuel)) \end{gathered}$$

The extension of our approach to other protocols and policies in the FIPA specification is fairly straightforward. Our approach shows how a well-defined normative concepts can be used to propose a high-level ACL pragmatics that are declarative, takes into account the context and that helps agents to achieve the perlocutionary effects of the speech acts. These two properties of the normative pragmatics, *contextual* and *perlocutionary*, fill in the last gaps in the list of requirements for ACLs discussed in section 1 and table 1. Next section offers a comparison to other approaches and discusses some short term future work necessary to improve the ongoing work presented in this paper. As a final note, the simplicity of the protocols and policies specified in this section was intentional. An important point for any future application of agent communication languages remains the proposal of high-level but simple ACL semantics and pragmatics.

8.6 Programs

Fagin *et al.* introduce a simple programming language which can be easily related to an Interpreted System [28]. Although the language is designed to express agents' knowledge, it can be adapted for its use in specifying norms of conversation. The basic standard program for agent i consists of statement of the form

```
case of
        if $t_1 \wedge k_1$ do $a_1$
        if $t_2 \wedge k_2$ do $a_2$
end case
```

where the t_i's are tests about some facts, k_i are knowledge test for agent i and a_i denote agent i's actions. We modify these knowledge-based programs to express tests over obligations, rights and permissions of agents, namely, to normative-based programs. The normative component consists of a Boolean combination of the form $O_i\varphi$ where φ can be an arbitrary formula that may include other deontic and temporal operators. Using this simple language we can express high-level protocols for agent communication. We represent the Fipa Request protocol specified above in table 9.

At first glance, it may seem a bit odd to use obligations after the operator **do**. However, in the interpretation of obligations and rigths provided by $NLTL_I$, $O_i\varphi$ means that "φ holds in agent i is working correctly" whereas $R_i\varphi$ is interpreted as "φ holds at some point of the system (r, m) if agent i is acting rightly".

```
case of
    if $(principal_i \wedge secretary_j) \wedge R_i(request(i, j, \phi))$ do $send_i(request(, j, \phi))$
    if $receive_j(request(i, j, \phi)) \wedge R_j(refuse(j, i, \phi))$ do $sent_j(refuse(j, i, \phi))$
    if $receive_j(request(i, j, \phi)) \wedge R_j(agree(j, i, \phi))$ do $sent_j(agree(j, i, \phi))$
    if $sent_j(agree(j, i, \phi)) \wedge F\phi$ do $O_j(inform(j, i, \phi))$
    if $sent_j(agree(j, i, \phi)) \wedge \neg F\phi$ do $O_j(failure(j, i, \phi))$
end case
```

Table 9. Program for Request Protocol.

Therefore, the last statement of the program denotes that if agent j has *agreed* to bring about some ϕ to agent j and ϕ does not eventually happens in the run of the system then agent j does send a failure message to agent i if working correctly.

9 Concluding Remarks

The characterization of roles is inspired by the work done on organizational concepts [53,45]. Other authors [18], have also presented temporal deontic logic with dynamic operators, but the combination of deontic, dynamic and temporal notions results in a logic that is too complex for our purposes.

In a very recent paper Boella *et al.* [54] present a role-based approach to ACL semantics. They intend to make the ACL semantics public by attributing mental states to social roles instead of agents. Thus, there are two sets of beliefs, those that are public and are ascribed to roles, and those that are private and belong to the agents' private mental states. A role is constrained by a set of social rules (rights, obligations, permissions, etc.) that define the expected behaviour of any agent playing the role. These social rules may or may not conflict the private beliefs and goals of agents. In any case, even if beliefs and goals are attributed to roles, agents playing a role would still need to reason about their beliefs and goals. From a semantic point of view, defining the ACL semantics in terms of roles makes the semantics less general, since the meaning of speech acts would be affected by agents' role. For example, two roles that are considered are those of *speaker* and *receiver*.

We believe that this paper offers a new framework for agent communication where the meaning of speech acts consists of the combination of the semantic specification and the NPRAG rules that constrain their use.

First, it clearly distinguishes semantics and pragmatics of the language. Semantically, it offers a computationally grounded specification language based on $MLTL_I$. This enables to define meaningful and public communicative actions. Regarding the pragmatics, it presents a procedure using normative rules to guide agents in conversation. Unlike research in ACL semantics, there are not many works that attempt to capture both aspects of communication in the same framework.

Considering the list of requirements for ACLs discussed, the approach presented in this paper achieves a number of objectives. After the semantics of the language was specified, the aim was to produce a pragmatic theory that would consider how contextual information constrains agents' behaviour, and how proposing normative rules for the use of speech acts facilitate the achievement of the perlocutionary effects.

1. **Autonomous**: The ACL semantics (SAL) do not completely fix agents communicative behaviour because the fulfilment of the perlocutionary effects are left to the ACL pragmatics.
2. **Complete**: We have defined a complete set of speech acts, understanding "complete" as representing every category in Searle's taxonomy. Searle's taxonomy is by no means a closed list; one could imagine a more fine-grained taxonomy including more systematic distinctions between types of directives such as yes/no questions, prohibitives, etc. However, this paper completes FIPA specification by defining speech acts for commissives and declaratives.
3. **Context**: In agent communication contextual factors include the role that agents play in the application scenario, the delegated tasks agents try to achieve, the propositional content of messages, and the record of previous exchanges. The use of normative concepts to model ACL pragmatics keep to a minimum agents' reasoning about each others' mental states. In that sense, it is more *efficient*. Furthermore, by avoiding that reasoning, the specification of conversation protocols and policies is greatly *simplified*.
4. **Declarative**: By providing a declarative definition of ACL semantics and pragmatics, specifying what the meaning is instead of a follow-the-rule low-level procedure, the resultant unified ACL is a high-level language.
5. **Formal**: The unified ACL is specified using two formal logics, $MLTL_I$ and $NLTL_I$ that describe the evolution of a multi-agent systems with respect to the agents' beliefs, goals, intentions, obligations and rights. A particular care was to provide an external interpretation of beliefs, goals and intentions in a way that those attitudes would refer states of a system instead of private mental states of the agents. In doing so, we were paving the ground provide a semantics and pragmatics suitable for verification.
6. **Grounded**: The notion of interpreted system was introduced [28] upon which the two specification languages $MLTL_I$ $NLTL_I$ were grounded.
7. **Public**: We claim that the illocutive/intentional aspect of communication should be preserved in the ACL semantics. This paper proposes an external interpretation of motivational concepts by relating them to states of agents in a system.
8. **Perlocutionary**: Conversation norms in the form of protocols and policies enable agents to achieve the perlocutionary effects by specifying obligations and rights on the participants. In order to preserve agents' autonomy, we offer a notion of right which specifies agents' behaviour when acting *rightly*.

It should be made clear that complying with these requirements is not the end of the story but rather its beginning. In other words, we see these properties as the starting point for the development of agent communication languages. A number of problems are still to be solved including issues of verification, implementation and the interaction of the communicative module with the rest of the social structure of the system.

Further work includes providing proofs for some properties of the interaction protocols with respect to interpreted systems. Furthermore, it would be interesting to provide a detail proof of the soundness and completeness of $MLTL_I$ and $NLTL_I$. We also need to verify the semantics of $NLTL_I$ in various ways. There are various methods of verification which depend on the type of ACL, on the information available, and on whether we are interested in verifying the ACL at design time or at run time [7]. Unlike other approaches, we are particularly interested in verifying the ACL pragmatics (only) because the pragmatics encodes the general communicative behaviour of agents. Following this, the type of the ACL to be verified corresponds, in our approach, to the normative component. Current work on pre-runtime verification of complex formal logics [55,56] looks very promising. Furthermore, it would be interesting to produce more sophisticated implementations of conversation protocols and policies in a manner that they could be integrated with platforms such as 3APL and BOID [57,45].

Deontic concepts are increasingly used in the specification and verification of multi-agent systems. It is unrealistic to assume that a whole open multi-agent system may be controlled by the same vendor. Thus, this makes it difficult to verify agents' conformance with the set of semantic and pragmatic specifications of ACLs. In this sense, by adopting a normative point of view, it seems more sensible to leave open the theoretical possibility of agents violating the norms. We can then use the formal language provided to reason about the consequences that result from those violations. Separating the specification language (from the implementation language) allows us to reason about external properties of the system. Further work on these issues would include the definition of more normative notions to complement *right* which may be more suitable to specific circumstances, and to embed our ACL in a normative multi-agent system.

References

1. FIPA ACL: FIPA Communicative Act Library Specification (2002) http://www.fipa.org/repository/aclspecs.html.
2. Finin, T., Weber, J.: Specification of the KQML Agent-Communication Language (1993) Finin T. and Weber J. Draft. Specification of the KQML Agent Communication Language. The DARPA Knowledge Sharing Initiative, External Interfaces Working Group.
3. Singh, M.P.: A social semantics for agent communication languages. In: Issues in Agent Communication, volume 1916 of LNAI. Berlin: Springer-Verlag (2000)
4. Colombetti, M., Formara, N., Verdicchio, M.: The role of institution in multi-agent systems. In: Atti del VII convegno dell'Associazione italiana per l'intelligenza artificiale (AI*IA 02). (2002)
5. Searle, J.R.: Speech Acts. An Essay in the Philosophy of Language. Cambridge: Cambridge University Press (1969)

6. Wooldridge, M.: Verifying that agents implement a communication language. In: Sixteenth National Conference on Artificial Intelligence (AAAI-00), Orlando, FL (1999)
7. Guerin, F., Pitt, J.: Agent Communication Frameworks and Verification. In: AAMAS 2002 Workshop on Agent Communication Languages, Bologna. (2002)
8. Chaib-Draa, B., Labrie, M.A., Bergeron, M., Pasquier, P.: Diagal: An agent communication language based on dialogue games sustained by social commiments. Automous Agents and Multi Agent Systems **13** (2006) 61–95
9. Flores, R.A., Kremer, R.C.: To commit or not to commit: Modelling agent conversations for action. Computational Intelligence **18** (2002)
10. Fornara, N., Colombetti, M.: A commitment-based approach to agent communication. Applied Artificial Intelligence an International Journal **18** (2004) 853–866
11. Agerri, R., Alonso, E.: A semantic and pragmatic framework for the specification of agent communication languages: Motivational attitudes and norms. In Kolp, M., Bresciani, P., Henderson-Sellers, B., Winikoff, M., eds.: Agent-Oriented Information Systems III (Post-Proceedings). Volume 3529., Heidelberg, Springer-Verlag (2006) 16–31
12. Cost, R.S., Chen, Y., Finin, T., Labrou, Y., Peng, Y.: Modeling agent conversations with colored petri nets. In: Workshop on Specifying and Implementing Conversation Policies, Third International Conference on Autonomous Agents (Agents '99), Seattle (1999)
13. Greaves, M., Holmback, H., Bradshaw, J.: What is a Conversation Policy? In Dignum, F., Greaves, M., eds.: Issues in Agent Communication. Heidelberg, Germany: Springer-Verlag (2000) 118–131
14. Pitt, J., Mamdani, A.: A protocol-based semantics for an agent communication language. In: Proceedings of the 16th International Joint Conference on Artificial Intelligence IJCAI'99, Stockholm, Morgan-Kaufmann Publishers (1999) 486–491
15. Phillips, L.R., Link, H.E.: The role of Conversation Policy in carrying out Agent Conversations. In: Workshop on Specifying and Implementing Conversation Policies, Autonomous Agents'99, Seattle. (1999)
16. Kagal, L., Finin, T.: Modelling conversation policies using permissions and obligations. Autonomous Agents and Multi Agent Systems (2007)
17. Holmback, H., Greaves, M., Bradshaw, J.: Agent A, Can you pass the salt? - the role of Pragmatics in Agent Communication (1999) http://citeseer.nj.nec.com/holmback98agent.html.
18. Dignum, F., Kuiper, R.: Combining dynamic deontic logic and temporal logic for the specification of deadlines. In Sprague, J.R., ed.: Proceedings of thirtieth HICSS, Hawaii (1997)
19. Esteva, M., Rodriguez, J.A., Sierra, C., Garcia, P., Arcos, J.L.: On the formal specification of electronic institutions. In Dignum, F., Sierra, C., eds.: Agent-mediated Electronic Commerce (The European AgentLink Perspective). Volume volume 1191 of LNAI. Springer, Berlin (2001) 126–147
20. Jones, A.J.I., Sergot, M.: On the characterisation of law and computer systems: The normative systems perspective. In Meyer, J.J., Wieringa, R., eds.: Deontic Logic in Computer Science: Normative System Specification. Wiley (1993)
21. Meyer, J.J., Wieringa, R., eds.: Deontic logic in computer science: normative system specification. Wiley (1993)
22. Norman, T.J., Sierra, C., Jennings, N.R.: Rights and commitment in multi-agent agreements. In: Proceedings of the Third International Conference on Multi-Agent Systems. (1998) 222–229

23. van der Torre, L.: Contextual deontic logic: Normative agents, violations and independence. Ann. Math. Artif. Intell. **37** (2003) 33–63
24. Fornara, N., Vigano, F., Colombetti, M.: Agent communication and institutional reality. In: AAMAS 2004 Workshop on Agent Communication (AC2004), New York (2004)
25. Chaib-Draa, B., Dignum, F.: Trends in agent comunication language. Computational Intelligence **18** (2002) 89–101
26. Wooldridge, M.: Semantic issues in the verification of agent communication languages. Journal of Autonomous Agents and Multi-Agent Systems **3** (2000) 9–31
27. Mayfield, J., Labrou, Y., Finin, T.: Evaluation of KQML as an Agent Communication Language. In Wooldridge, M., Muller, J.P., Tambe, M., eds.: Lecture notes in Artificial Intelligence. Berlin: Springer-Verlag (1996) 1–12
28. Fagin, R., Halpern, J.Y., Moses, Y., Vardi, M.Y.: Reasoning about Knowledge. The MIT Press, Cambridge, MA (1995)
29. van der Hoek, W., Wooldridge, M.: Towards a logic of rational agency. Logic Journal of the IGPL **11** (2003) 135–159
30. Lomuscio, A., Sergot, M.: Deontic interpreted systems. Studia Logica **75** (2003)
31. Wooldridge, M.: Computationally grounded theories of agency. In Durfee, E., ed.: Proceedings of the Fourth International Conference on Multi-Agent Systems (ICMAS 2000), IEEE Press (2000)
32. Kripke, S.: Semantical considerations on modal logic. Acta Philosophical Fennica **XVI** (1963) 83–94
33. Cohen, P., Levesque, H.: Communicative actions for artificial agents. In Bradshaw, J.M., ed.: Software Agents. AAAI Press / The MIT Press, Cambridge (MA) (1997)
34. Rao, A.S., Georgeff, M.P.: Decision procedures for bdi logics. Journal of Logic and Computation **8** (1998) 293–342
35. Halpern, J., Vardi, M.: The complexity of reasoning about knowledge and time.i. lower bounds. Journal of Computer and System Sciences **38** (1989) 195–237
36. van der Meyde, R., Wong, K.: Complete axiomatizations for reasoning about knowledge and branching time. Studia Logica **75** (2003) 93–123
37. Halpern, J., van der Meyden, R., Vardi, M.: Complete axiomatizations for reasoning about knowledge and time. SIAM Journal of Computing **33** (2004) 449–478
38. Halpern, J.Y., Vardi, M.Y.: The complexity of reasoning about knowledge and time: synchronous systems. Technical Report RJ 6097, IBM (1988)
39. Lomuscio, A., Wozna, B.: A complete and decidable axiomatization for deontic interpreted systems. In: 8th International Workshop on application of Deontic Logic to Computer Science (DEON-06). LNAI, Heidelberg, Springer-Verlag (2006)
40. Manna, Z., Pnueli, A.: The Temporal Logic of Reactive and Concurrent Systems. Springer-Verlag, Berlin, Germany (1995)
41. Agerri, R.: Motivational Attitudes and Norms in a unified Agent Communication Language for open Multi-Agent Systems: A Pragmatic Approach. PhD thesis, Dept. of Computing, City University, London, UK (2006)
42. Rao, A., Georgeff, M.: Modeling rational agents within a bdi-architecture. In Allen, J., Fikes, R., Sandewall, E., eds.: 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR'91), Morgan Kaufmann Publishers (1991) 473–484
43. Austin, J.L.: How to do Things with Words. Oxford University Press, Oxford (1962)
44. Alonso, E.: Rights and argumentation in open multi-agent systems. Artificial Intelligence Review **21** (2004) 3–24

45. van der Torre, L., Hulstijn, J., Dastani, M., Broersen, J.: Specifying multiagent organizations. In: Proceedings of the Seventh Workshop on Deontic Logic in Computer Science (Deon'2004), LNAI 3065. Springer (2004) 243–257
46. von Wright, G.H.: Deontic logic. Mind **60** (1951) 1–15
47. Boella, G., van der Torre, L.: Role-based rights in artificial social systems. In: Proceedings of IAT05. IEEE, 2005. (2005)
48. Castelfranchi, C.: Practical permission: Dependence, power and social commitment. In: Proceedings of 2nd workshop on Practical Reasoning and Rationality, London (1997)
49. Wenar, L.: Rights. In Zalta, E.N., ed.: The Stanford Encyclopedia of Philosophy. http://plato.stanford.edu/archives/win2005/entries/rights/ (Winter 2005)
50. Anderson, A.R.: Some nasty problems in the formalization of ethics. Nous **1** (1967) 345–360
51. Endriss, U.: Temporal logic for representing agent communication protocols. In: Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2005), ACM Pres (2005)
52. Endriss, U., Maudet, N., Sadri, F., Toni, F.: Logic-based communication protocols. In: Advances in Agent Communication, LNAI, volume 2922, pages 91-107, Springer-Verlag. (2004)
53. Ferber, J., Gutknecht, O.: A meta-model for the analysis of organizations in multiagent systems. In: Proceedings of the Third International Conference on Multi-Agent Systems (ICMAS'98). (1998) 128–135
54. Boella, G., Damiano, R., Hulstijn, J., van der Torre, L.: Acl semantics between social commitments and mental attitudes. In: In proceedings of the AC 2006. (2006)
55. Raimondi, F., Lomuscio, A.: Automatic verification of deontic properties of multiagent systems. In: Deontic Logic. Volume LNCS 3065., Springer Verlag (2004) 228–242
56. Baldoni, M., Baroglio, C., Martelli, A., Patti, V.: Verification of protocol conformance and agent interoperability. In Toni, F., Torroni, P., eds.: Sixth International Workshop on Computational Logic in Multi-Agent Systems. Volume 3900 of LNCS., Heidelberg, Springer (2006) 265–283
57. Dastani, M., Riemsdijk, M., Dignum, F., Meyer, J.J.: A programming language for cognitive agents: Goal-directed 3apl. In Dastani, M., Dix, J., Fallah-Seghrouchni, A.E., eds.: Programming Multi-Agent Systems (LNAI 3037). Springer-Verlag, Berlin (2004) 111–130