

A Logical Model of Social Commitment for Agent Communication

Mario Verdicchio
Politecnico di Milano

Dipartimento di Elettronica e Informazione
Piazza Leonardo da Vinci 32, 20133 Milano, Italy
Phone: +39-02-2399-3686

Mario.Verdicchio@Elet.PoliMi.It

Marco Colombetti
Politecnico di Milano

Piazza Leonardo da Vinci 32, 20133 Milano, Italy
University of Lugano
Via Buffi 13, 6904 Lugano, Switzerland

Marco.Colombetti@PoliMi.It

ABSTRACT

As part of the goal of developing a genuinely open multiagent system, many efforts are devoted to the definition of a standard Agent Communication Language (ACL). The aim of this paper is to propose a logical framework for the definition of ACL semantics based upon the concept of (social) commitment. Our framework relies on the assumption that agent communication should be analyzed in terms of communicative acts, by means of which agents create and manipulate commitments, provided certain contextual conditions hold. We propose formal definitions of such actions in the context of a temporal logic that extends CTL* with past-directed temporal operators. In the system we propose, called CTL[±], time is assumed to be discrete, with no start or end point, and branching in the future. CTL[±] is then extended to represent actions and commitments; in particular, we formally define the conditions under which a commitment is fulfilled or violated. Finally, we show how our logic of commitment can be used to define the semantics of an ACL.

Categories and Subject Descriptors

I.2.0 [Artificial Intelligence]: General – Philosophical foundations. I.2.11 [Artificial Intelligence]: Distributed Artificial Intelligence – Multiagent systems.

General Terms

Languages, Theory.

Keywords

Agent Communication Languages, Commitment, Speech Act Theory, Temporal Logic.

1. INTRODUCTION

One of the main goals in the field of autonomous agents is the development of genuinely open multiagent systems. As part of this enterprise, many efforts are devoted to the definition of a

standard Agent Communication Language (ACL). So far, two ACLs have been widely discussed in the literature: KQML [4] and FIPA ACL [5], but we do not yet have a universally accepted standard. In particular, there is no general agreement on the definition of ACL semantics.

The aim of this paper is to propose a framework for the definition of ACL semantics based upon the concept of (social) commitment, thus adopting an approach that has already been proposed and discussed by some authors [10,2]. In our view, a commitment-based approach to semantics has remarkable advantages over the more traditional proposals based on mental states (see for example [1,7]). In particular, commitments, contrary to mental states, are public and observable, thus they do not need to be attributed to other agents by means of inference processes, and can be stored in public records for further reference.

Our framework, like all major proposals in the field of ACLs, relies on the assumption that agent communication should be analyzed in terms of communicative acts. In our view, communicative acts are performed by agents to create and manipulate commitments. That is, agents modify the social state of a multiagent system by carrying out speech acts that affect the network of commitments binding agents to one another. For instance, when agent *a* informs agent *b* that *p*, then *a* becomes committed, relative to *b*, to the fact that *p* holds. As we shall show in the rest of this paper, we can similarly model other kinds of communicative acts from the perspective of commitments.

Previous versions of our model have been published elsewhere [2,6]. However, we try here for the first time to delineate a full logical model of commitment, including the aspects related to time. In Section 2 we illustrate some aspects of our model of time and action. In Section 3 we present a formal model of commitment. In Section 4 we investigate on the relations between message exchanges, communicative acts and the creation and manipulation of commitments. Finally, in Section 5, we draw our conclusions and illustrate some future work.

2. TIME AND ACTION

2.1 Time

For the treatment of time, we adopt a framework close to the CTL* temporal logic [3]. As is well known, CTL* is a powerful logic of branching time, developed to prove properties of computational processes. In the context of agent interaction, we found it necessary to extend CTL* with past-directed temporal operators. In the system we propose, called CTL[±], time is assumed to be discrete, with no start or end point, and branching in the future.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

AAMAS '03, July 14-18, 2003, Melbourne, Australia.
Copyright 2003 ACM 1-58113-683-8/03/0007...\$5.00.

The formal language \mathbf{L} of CTL^\pm is the smallest set such that:

- $\mathbf{A} \subseteq \mathbf{L}$, where \mathbf{A} is a suitable set of atomic formulae;
- $\neg \mathbf{L} \subseteq \mathbf{L}$, $(\mathbf{L} \wedge \mathbf{L}) \subseteq \mathbf{L}$;
- $\mathbf{X}^+ \mathbf{L} \subseteq \mathbf{L}$, $\mathbf{X}^- \mathbf{L} \subseteq \mathbf{L}$, $(\mathbf{L} \mathbf{U}^+ \mathbf{L}) \subseteq \mathbf{L}$, $(\mathbf{L} \mathbf{U}^- \mathbf{L}) \subseteq \mathbf{L}$;
- $\mathbf{A} \mathbf{L} \subseteq \mathbf{L}$, $\mathbf{E} \mathbf{L} \subseteq \mathbf{L}$.

The intuitive meaning of the temporal operators is the same as in CTL^* , with the additional stipulation that:

- \mathbf{X}^+ means *at the next instant* (in the future);
- \mathbf{X}^- means *at the previous instant* (in the past);
- \mathbf{U}^+ means *until* (in the future);
- \mathbf{U}^- means *since* (in the past).

\mathbf{A} and \mathbf{E} are *path quantifiers*, respectively meaning *for all paths* and *for some path*.

To define the formal semantics of \mathbf{L} , let S be a set of *states*. A CTL^\pm *frame* F on S is an infinite tree-like structure on S , where every state has exactly one *predecessor* and a nonempty set of *successors*.

A *path* in frame F is an infinite sequence $p = \langle p_0, \dots, p_n, \dots \rangle$ of states, such that for every element p_n in the sequence, element p_{n+1} is one of the successors of p_n in F . The subsequence of p starting from element p_n is itself a path, and will be denoted by p^n . The set of all paths starting from state s will be denoted by $\text{Paths}(s)$. Paths allow us to formalize the concepts of being “in the past” or “in the future” of some state. More precisely, we say that state s' is *in the future of* s (in frame F) iff there is a path p such that $s = p_0$ and, for some n , $s' = p_n$. Symmetrically, we say that s' is *in the past of* s (in frame F) iff there is a path p such that $s' = p_0$ and, for some n , $s = p_n$.

A CTL^\pm *model* is a pair $M = \langle F, v \rangle$, where F is a CTL^\pm frame and v is an evaluation function assigning a Boolean truth value to every atomic formula at every state. We are now ready to define the truth conditions of an \mathbf{L} formula in model M on path p :

- $M, p \models \varphi$, where φ is an atomic formula, iff $v(\varphi, p_0) = 1$;
- $M, p \models \neg \varphi$ iff $M, p \not\models \varphi$;
- $M, p \models (\varphi \wedge \psi)$ iff $M, p \models \varphi$ and $M, p \models \psi$;
- $M, p \models \mathbf{X}^+ \varphi$ iff $M, p^1 \models \varphi$;
- $M, p \models \mathbf{X}^- \varphi$ iff for some path q , ($q^1 = p$ and $M, q \models \varphi$);
- $M, p \models (\varphi \mathbf{U}^+ \psi)$ iff for some n , ($M, p^n \models \psi$ and for all m s.t. $0 \leq m < n$, $M, p^m \models \varphi$);
- $M, p \models (\varphi \mathbf{U}^- \psi)$ iff for some path q and for some n , ($q^n = p$ and $M, q \models \psi$ and for all m s.t. $0 \leq m < n$, $M, q^m \models \varphi$);
- $M, p \models \mathbf{A} \varphi$ iff for all $q \in \text{Paths}(p_0)$, $M, q \models \varphi$;
- $M, p \models \mathbf{E} \varphi$ iff for some $q \in \text{Paths}(p_0)$, $M, q \models \varphi$.

We define an \mathbf{L} formula to be *true in model* M at state s iff it is true in M on all paths starting from s :

- $M, s \models \varphi$ iff for all $p \in \text{Paths}(s)$, $M, p \models \varphi$.

Finally, we define a formula to be *valid* iff it is true on all paths of every model:

- $\models \varphi$ iff for all M and all p , $M, p \models \varphi$.

Taking the temporal operators \mathbf{X}^+ , \mathbf{X}^- , \mathbf{U}^+ , and \mathbf{U}^- as primitives, we can introduce the following operators as abbreviations: \mathbf{F}^+ (sometimes in the future), \mathbf{F}^- (sometimes in the past), \mathbf{G}^+ (always in the future), \mathbf{G}^- (always in the past):

- $\mathbf{F}^+ \varphi =_{\text{def}} \text{true} \mathbf{U}^+ \varphi$,
- $\mathbf{F}^- \varphi =_{\text{def}} \text{true} \mathbf{U}^- \varphi$,
- $\mathbf{G}^+ \varphi =_{\text{def}} \neg \mathbf{F}^+ \neg \varphi$,
- $\mathbf{G}^- \varphi =_{\text{def}} \neg \mathbf{F}^- \neg \varphi$.

We also define a “weak until” and a “weak since” temporal operators:

- $\varphi \mathbf{W}^+ \psi =_{\text{def}} \mathbf{G}^+ \varphi \vee \varphi \mathbf{U}^+ \psi$,
- $\varphi \mathbf{W}^- \psi =_{\text{def}} \mathbf{G}^- \varphi \vee \varphi \mathbf{U}^- \psi$.

Later on we shall use another derived operator, representing the intuitive concept of “until-and-no-longer”. This operator is defined as follows:

- $\varphi \mathbf{Z}^+ \psi =_{\text{def}} \varphi \mathbf{W}^+ \psi \wedge \mathbf{G}^+(\psi \rightarrow \mathbf{G}^+ \neg \varphi)$.

In other words, $\varphi \mathbf{Z}^+ \psi$ is true iff in the future: ψ never becomes true and φ is true forever, or ψ eventually becomes true and since then φ is no longer true. More derived temporal operators will be defined later on to treat specific examples.

2.2 Events and Actions

We now extend the temporal language \mathbf{L} of CTL^\pm in order to represent events and actions. We do this by introducing a number of predicates on sorted arguments.

We reify events, that is, we treat them as a sort of individuals, called *event tokens*. Every event token belongs to at least one *event type*, and takes place (*happens*) at exactly one time instant. We focus on a special kind of events, *actions*, which are brought about by an agent, called the *actor* of the action.

In the following, variables e, e', \dots , will range on event tokens; variables x, y, \dots , will range on agents; and variables t, t', \dots , will range on event types. We take $\text{Happ}(e)$, $\text{Type}(e, t)$ and $\text{Actor}(e, x)$ as primitive predicates, and define:

- $\text{Done}(e, x, t) =_{\text{def}} \text{Happ}(e) \wedge \text{Type}(e, t) \wedge \text{Actor}(e, x)$.

The formula $\text{Done}(e, x, t)$ expresses the fact that event e of type t is brought about by agent x . For the sake of convenience, at times we shall use the “m-dash” character to express existential quantification, as in the example below:

- $\text{Done}(e, -, t) =_{\text{def}} \exists x \text{Done}(e, x, t)$.

The semantics of \mathbf{L} has now to be enriched to account for the interpretation of the extended language. This can be done by: adding a typed domain D of individuals to every model M ; defining an interpretation of first-order terms into D ; defining an interpretation of primitive predicates in D ; and defining the semantics of the first-order quantifiers \forall and \exists . In this paper we do not develop these technical aspects in details, and thus rely on the reader’s intuition for the interpretation of first-order expressions.

As usual, we now need to introduce a number of axioms to constrain the interpretation of primitive predicates. It should be noted that such axioms do not alter the structure of temporal frames, but reduce the set of allowable models by putting constraints on the interpretation of terms and predicates. Validity of formulae must then be understood with respect to the class of CTL^\pm models that satisfy such constraints.

As we already said, the instant at which an event takes place on a path is unique. We therefore adopt the axiom

- (UH) $\text{Happ}(e) \rightarrow \mathbf{X}^- \mathbf{G}^- \neg \text{Happ}(e) \wedge \mathbf{A} \mathbf{X}^+ \mathbf{G}^+ \neg \text{Happ}(e)$.

3. COMMITMENT

3.1 Representing Commitments

We define a commitment as a social state between agents including three components:

- the *debtor*, that is, the agent that is committed;
- the *creditor*, that is, the agent relative to which the debtor is committed,
- the *content*, that is, the state of affairs to which the debtor is committed relative to the creditor.

A commitment is said to be a *precommitment* when it has been proposed, but not yet accepted or refused. In such a case, we say that the (potential) debtor is precommitted to the (potential) creditor. In our treatment, both precommitments and actual commitments arise from the performance of communicative acts.

We view (social) commitment as a deontic state, akin to obligation. For such a reason, it is essential to define when a commitment is *fulfilled* and when it is *violated*. We shall give the relevant formal definitions in the following subsections. However, in this paper we do not investigate what is going to happen when a commitment is fulfilled or violated (e.g., in terms of agent reputation, sanctions, etc.). These are important aspects of multiagent systems management, but go beyond the conceptual definition of commitment.

We now extend our formal language to accommodate for the treatment of commitments. The resulting language will be called *Semantic Language*, given that its purpose is to define the semantics of ACL messages. To represent a commitment, we need to represent a debtor, a creditor, and a content. Debtors and creditors are agents, and shall be represented by first-order terms of sort *agent* like we already did in Subsection 2.2. The representation of content is more critical. It seems to us that there are basically two possibilities:

- The content can be represented by a formula of the Semantic Language. In this case, commitment can be represented through a modal operator, analogously to the deontic logic representation of obligation.
- The content can be represented as a first-order term. In this case, a commitment can be represented by a first-order formula.

We believe there are at least two reasons to adopt the latter solution. The first, obvious reason is that the technicalities required by a predicative representation are simpler than the ones required by a modal representation. The second, more important, reason is that in the context of agent communication the content of a commitment, as we shall see later on, is always derived from an agent message. More precisely, a commitment's content derives from a statement in some Content Language (CL): think for example of the value of the `:content` parameter in KQML or FIPA ACL messages. With respect to a CL, the Semantic Language we are presently defining can be viewed as a meta-language. It is therefore feasible to represent a CL statement by a first-order term of the Semantic Language. Such a first-order term may be viewed as the representation of the abstract syntax of a concrete CL statement. Of course, in the Semantic Language it is not sufficient to represent the syntax of a CL statement: we also need to represent its semantics. To do so, we shall assume that:

- The abstract syntax of any CL statement can be represented by a first-order term of the Semantic Language.

- If u is such a term, then the meaning of the corresponding statement is represented by a formula of the Semantic Language, which we shall denote by $\lfloor u \rfloor$. In other words, $\lfloor u \rfloor$ is a truth-preserving translation of u into a formula of the Semantic Language. For such a translation to be possible the Semantic Language will have to include enough predicate, function, and constant symbols to represent the meaning of CL statements.

We introduce two predicates, *Comm* and *Prec*, to represent commitments and precommitments. In particular,

$Comm(e,x,y,u)$

will mean that event e has brought about a commitment for agent x , relative to agent y , to the truth of $\lfloor u \rfloor$. When the above formula is true, we shall say that e is a *commitment-inducing event*. Precommitments are represented analogously:

$Prec(e,x,y,u)$

will mean that event e has brought about a precommitment for agent x , relative to agent y , to the truth of $\lfloor u \rfloor$.

Under given conditions, that we shall analyze later on, commitments can be *made* or *cancelled*, and precommitments can be *made*, *cancelled* or *accepted* (i.e., turned into actual commitments). This is possible thanks to the performance of tokens of suitable action types, formally defined in the next subsection: make commitment (*mc*), make precommitment (*mp*), cancel commitment (*cc*), cancel precommitment (*cp*), and accept precommitment (*ap*). Such actions, as we shall see later on, are performed by exchanging messages in an ACL.

3.2 A Logical Model of Commitment

The action types for commitment manipulation are defined by axioms describing their *constitutive effects*, that is, by describing the state of affairs that necessarily hold if a token of a given action type is successfully performed.

Make Commitment

$$(MC) \quad Done(e,-,mc(x,y,u)) \rightarrow A (Comm(e,x,y,u) Z^+ Done(-,-,cc(e,x,y,u))).$$

Axiom MC says that:

if an agent (not necessarily x or y) successfully performs an action of making a commitment with x as the debtor, y as the creditor, and u as the content,

then on all paths x is committed, relative to y , to content u ,

until an agent possibly cancels such a commitment, after which the commitment no longer exists.

It is important to remark that Axiom MC only defines what making a commitment means. It does not establish in what way, and under what conditions, an agent may actually make or cancel a commitment in a concrete situation. This aspect will be dealt with in Section 4.

Make Precommitment

$$(MP) \quad Done(e,-,mp(x,y,u)) \rightarrow A (Prec(e,x,y,u) Z^+ (Done(-,-,ap(e,x,y,u)) \vee Done(-,-,cp(e,x,y,u)))).$$

Axiom MP is analogous to MC.

Accept Precommitment

$$(AP) \quad Done(e',-,ap(e,x,y,u)) \wedge \neg Done(-,-,cp(e,x,y,u)) \rightarrow A (Comm(e',x,y,u) Z^+ Done(-,-,cc(e',x,y,u))).$$

Axioms AP says that:

if an agent successfully performs an action of accepting a precommitment brought about by event e , with debtor x , creditor y , and content u ,

and no agent has just cancelled such a precommitment,

then the action of acceptance brings about on all paths a commitment for x , relative to y , to content u , which will stand until it is possibly cancelled.

Again, this axiom does not say by what means or under what conditions an agent may actually accept a precommitment in a concrete situation.

The next axiom assures that an event, which takes place at a certain instant, can (pre)commit agents only from that moment on. In other words, no (pre)commitment is retroactive:

$$Happ(e) \rightarrow XG^-(\neg Prec(e,x,y,u) \wedge \neg Comm(e,x,y,u)).$$

Finally, the next axiom states that all (pre)commitments are necessarily brought about by some event:

$$Prec(e,x,y,u) \vee Comm(e,x,y,u) \rightarrow F^- Happ(e).$$

3.3 Fulfillment and Violation

Intuitively, a commitment is fulfilled when its content is true, and is violated when its content is false. However, given that we are working in the context of branching-time logic, the formal definitions are not trivial.

Let us start with an informal example. Suppose that thanks to event e_1 , agent a is committed, relative to agent b , to the content expressed by CL sentence u_1 , whose intuitive meaning is “it will rain until midnight.” Suppose further that e_1 takes place at 4:00 pm, and that it persistently rains from that time to 6:00 pm, inclusive. Intuitively, at 6:00 the commitment induced by e_1 is neither fulfilled nor violated (we shall say that the commitment is *pending*). Now consider two possible developments:

- It goes on raining until midnight. In this case, the commitment induced by e_1 is fulfilled at time 0:00 am.
- At 6:01 pm it suddenly stops raining. In this case, the commitment induced by e_1 is violated at 6:01 pm.

In order to formalize these intuitions, two problems must be solved. The first problem has to do with the *temporal indexicality* of content sentences. By this we mean that the truth of the sentence “it will rain until midnight” has to be evaluated with respect to the state at which the commitment is made (the *point of speech*, in Reichenbach’s terminology¹ [9]). On the other hand, to know whether the commitment is fulfilled or violated we have to wait until something relevant happens, that is, until the first state at which it stops raining, or the first state at which it is midnight (Reichenbach’s *point of event*). But then, and this is the second problem, what is the truth value of the content at a generic state (Reichenbach’s *point of reference*) lying between the point of speech and the point of event?

We propose a solution in which:

- content sentences are temporally de-indexicalized in a simple and uniform way, by conjoining their translation into the Semantic Language with an atomic formula setting the point of speech;

¹ The German philosopher Hans Reichenbach proposed a famous model of verb tenses in Chapter 7 of his book *Elements of Symbolic Logic*. We adopt his terminology, but reinterpret it with some freedom.

- the truth value of a content sentence at a given point of reference is evaluated with respect to all paths starting from the point of reference.

Fulfillment

On the basis of our previous considerations, fulfillment can be formally defined as follows:

$$(FC) \text{ Fulf}(e,x,y,u) =_{\text{def}} Comm(e,x,y,u) \wedge AF^-(Happ(e) \wedge \lfloor u \rfloor).$$

To understand this definition correctly, it is helpful to go back to our previous example. Let us assume that

$$\lfloor u_1 \rfloor = (rain \ U^+ \ \text{midnight}),$$

and suppose that the commitment-inducing event e_1 takes place in model M at state s :

$$M,s \models Happ(e_1),$$

$$M,s \models A(Comm(e_1,a,b,u_1) \ Z^+ \ Done(\neg, -, cc(e_1,a,b,u_1))).$$

Now consider an arbitrary state s' in the future of s . We have

$$M,s' \models \text{Fulf}(e_1,a,b,u_1)$$

$$\text{iff } M,s' \models Comm(e_1,a,b,u_1) \wedge AF^-(Happ(e_1) \wedge \lfloor u_1 \rfloor).$$

Let us assume that the commitment made at s has not been cancelled until s' (inclusive). This implies that

$$M,s' \models Comm(e_1,a,b,u_1).$$

Under such conditions, the commitment is fulfilled at s' iff

$$M,s' \models AF^-(Happ(e_1) \wedge \lfloor u_1 \rfloor),$$

that is, iff for all $p \in Paths(s')$,

$$M,p \models F^-(Happ(e_1) \wedge \lfloor u_1 \rfloor).$$

Therefore, for the commitment to be fulfilled at s' , the formula

$$Happ(e_1) \wedge (rain \ U^+ \ \text{midnight})$$

must be true at some state in the past of s' . Now, thanks to Axiom UH (Section 2.2) we know that on every path the state at which an event takes place is unique. Thus, for the commitment to be fulfilled at s' , the formula

$$(rain \ U^+ \ \text{midnight})$$

must be true, for all $p \in Paths(s)$ going through s' . A model satisfying these requirements is depicted in Figure 1.

This example shows how statement u_1 is de-indexicalized by evaluating it in the state s at which event e_1 took place.

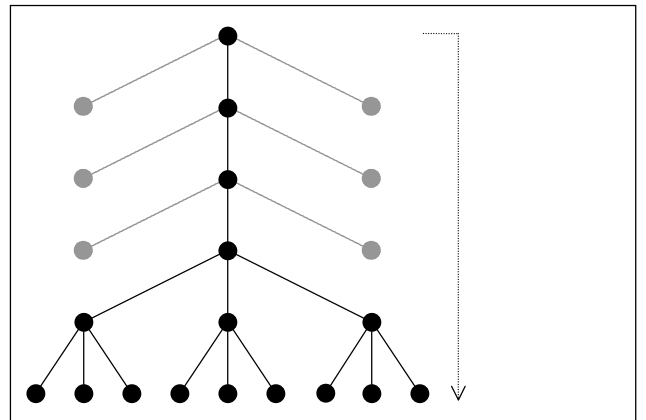


Figure 1. Formula $rain \ U^+ \ \text{midnight}$ is true on all paths starting from s and going through s' .

Moreover, the definition of fulfillment at s' considers the truth value of s_1 on all paths starting from s and going through s' . The set of such paths typically becomes smaller when s' is moved further in the future of s . As a consequence, a commitment that is not yet fulfilled at s may be fulfilled at some state s' in the future of s .

Violation

Analogously to fulfillment, we can define violation as follows:

$$(VC) \text{Viol}(e,x,y,u) =_{\text{def}} \text{Comm}(e,x,y,u) \wedge \text{AF}^-(\text{Happ}(e) \wedge \neg \lfloor u \rfloor).$$

Pending commitments

A commitment is pending iff it is neither fulfilled nor violated:

$$(PC) \text{Pend}(e,x,y,u) =_{\text{def}} \text{Comm}(e,x,y,u) \wedge \neg \text{Fulf}(e,x,y,u) \wedge \neg \text{Viol}(e,x,y,u).$$

Thanks to Axiom UH (Section 2.2), from the above definition we can derive:

$$\begin{aligned} \models \text{Pend}(e,x,y,u) \leftrightarrow & \text{Comm}(e,x,y,u) \\ & \wedge \text{EF}^-(\text{Happ}(e) \wedge \lfloor u \rfloor) \\ & \wedge \text{EF}^-(\text{Happ}(e) \wedge \neg \lfloor u \rfloor). \end{aligned}$$

However, Definition PC raises a fairly subtle formal problem, which we shall analyze in the next subsection.

3.4 Some Properties of Commitment

We shall now try to show that the axioms and definitions given in the previous subsections determine a satisfactory “logic of commitment.”

Let us start with a few notes on fulfillment and violation. It is easy to see that if a commitment is introduced through a *make commitment* or *accept precommitment* action and later cancelled, it can no longer be fulfilled or violated. This is a direct consequence of Axioms MC and AP, and of Definitions FC and VC. Even if a commitment has already been fulfilled or violated in the past, it is no longer fulfilled or violated after it is cancelled. It is possible, however, to express the idea that a commitment has been fulfilled or violated in the past, by using the F^- operator. It would also be possible to constrain *cancel commitment* actions so that commitments that have already been fulfilled or violated can no longer be cancelled.

Some commitments can be fulfilled, but can never be violated in a finite period of time. An example is a commitment whose content, translated into the Semantic Language, is F^+rain . Analogously, some commitments can be violated but never fulfilled in finite time. Consider for example a commitment to G^+rain .

All commitments whose content is logically valid are immediately fulfilled. Dually, all commitments whose content is logically contradictory are immediately violated. Moreover, all commitments whose point of event is in the past of the point of speech are immediately fulfilled or violated.

From the definitions of Section 3.3, every commitment is either fulfilled, or violated, or pending, and these three states are mutually exclusive. In fact it is possible to prove that

$$\begin{aligned} \models & \text{Comm}(e,x,y,u) \\ \rightarrow & \mathbf{xor}(\text{Fulf}(e,x,y,u), \text{Viol}(e,x,y,u), \text{Pend}(e,x,y,u)); \end{aligned}$$

that is, exactly one of $\text{Fulf}(e,x,y,u)$, $\text{Viol}(e,x,y,u)$, or $\text{Pend}(e,x,y,u)$ is true in all models at every state at which $\text{Comm}(e,x,y,u)$ holds. This result, however, should be interpreted with some care. To show why, let us go back again to our example of Section 3.3.

Suppose that thanks to event e_1 , agent a is committed, relative to agent b , to the fact that it will rain until midnight; that e_1 takes place at 4:00 pm; and that it persistently rains from that time to 6:00 pm, inclusive. As we have remarked in the previous subsection, at 6:00 the commitment induced by e_1 is intuitively *pending*. However, without further assumptions it is not possible to prove this. The reason is that there are models of the Semantic Language in which the commitment is not pending, but fulfilled. Consider for example a one-path frame, and assume that the atomic formula *rain* is true at every state. In such a model, the commitment to the fact that it will rain until midnight is fulfilled as soon as it is made. Given that in some models the commitment is fulfilled, it is not possible to prove that it is pending.

The problem has nothing to do with our definitions. Rather, it derives from the fact that certain intuitions about the world are not represented in the Semantic Language. In this case, the intuition is that rain is *contingent*, in the sense that it is always logically possible that it rains or that it does not rain at the next state. If we want to carry this intuition into the Semantic Language, we need to exclude all models that do not meet it. This can be done by assuming the following *contingency axiom* for rain:

$$\text{EX}^+rain \wedge \text{EX}^+\neg rain.$$

Of course, this axiom does not belong to a logical model of commitment, but represents a fragment of domain knowledge. Such knowledge has to be expressed in terms of suitable axioms if we want to derive properties of commitments that square with our intuitions about the world.

So far we said nothing about the behaviour of the commitment predicate with respect to the structure of content. To do so, however, we must make some assumptions about the abstract syntax of the CL. Let us assume that the CL allows for the Boolean connectives, quantifiers and temporal operators that, for the sake of simplicity, we will represent by the same symbols we use in the Semantic Language.

Now consider the formula

$$\text{Comm}(e,x,y, u \wedge v).$$

The question is: if e commits x , relative to y , to $u \wedge v$, does it also separately commit x to u and to v ? In fact, our logic does not allow us to derive $\text{Comm}(e,x,y,u)$ or $\text{Comm}(e,x,y,v)$ from $\text{Comm}(e,x,y, u \wedge v)$. It turns out, however, that we do not need to add anything to our axioms and definitions to obtain a satisfactory behavior of commitment with respect to conjunction. Indeed, it is easy to see that

$$\begin{aligned} \models & \text{Comm}(e,x,y, u \wedge v) \wedge \text{AF}^-(\text{Happ}(e) \wedge \neg \lfloor u \rfloor) \\ & \rightarrow \text{Viol}(e,x,y, u \wedge v), \\ \models & \text{Comm}(e,x,y, u \wedge v) \wedge \text{AF}^-(\text{Happ}(e) \wedge \neg \lfloor v \rfloor) \\ & \rightarrow \text{Viol}(e,x,y, u \wedge v). \end{aligned}$$

The validity of these formulae allows one to say, in informal speech, that

if a debtor is committed, relative to a creditor, to the conjunction of u and v ,

then the debtor is committed, relative to the creditor, to both u and v ,

in the sense that the falsity of either u or v implies a violation of the original commitment.

Another interesting problem is given by the treatment of *conditional commitments*, that is, commitments that become active provided some condition holds. Conditional commitments

are not trivial to define in terms of the material conditional, and are often given an *ad hoc* treatment (see for example [6,11]), not dissimilar from the treatment of conditional obligation in deontic logic. To see where difficulties come from, let us see what happens if a conditional commitment is simply defined as a commitment with a conditional content. Suppose for example that event e_1 commits agent a , relative to agent b , to the fact that if a lightning is seen, a thunder will be heard immediately after. Further suppose that event e_1 takes place in model M at state s (i.e., s is the point of speech), and let formula

$$(1) \text{Comm}(e_1, a, b, \text{lightning} \rightarrow X^+ \text{thunder})$$

express such a commitment. The obvious problem is that the commitment expressed by Formula 1 is immediately fulfilled if no lightning is seen at the point of speech, because

$$\text{AF}^-(\text{Happ}(e_1) \wedge (\text{lightning} \rightarrow X^+ \text{thunder}))$$

is true at s if *lightning* is false at s . This problem, however, is not due to a limitation of material conditional, but to the fact that Formula 1 does not correctly represent the content of the commitment. In fact, the statement to which a commits may be interpreted in two ways: (i), “always in the future, a thunder will be heard immediately after a lightning is seen;” or (ii), “as soon as a lightning will be seen, a thunder will be heard immediately after.” With the first interpretation, a ’s commitment is represented by

$$(2) \text{Comm}(e_1, a, b, G^+(\text{lightning} \rightarrow X^+ \text{thunder})).$$

The commitment of Formula 2 can never be fulfilled in finite time, and is violated at state s' , in the future of s , iff

$$\text{AF}^-(\text{Happ}(e_1) \wedge \neg G^+(\text{lightning} \rightarrow X^+ \text{thunder}))$$

holds at s' , that is, iff

$$\text{AF}^-(\text{Happ}(e_1) \wedge F^+(\text{lightning} \wedge \neg X^+ \text{thunder}))$$

holds at s' . In other words, the commitment of Formula 2 is violated in the future of s as soon as on all paths starting from s and going through the current state it is the case that a lightning will be seen that is not immediately followed by a thunder.

With the second interpretation, the commitment is expressed by

$$(3) \text{Comm}(e_1, a, b, \text{lightning } S^+ X^+ \text{thunder}),$$

where the “as soon as” operator S^+ is defined as below:

$$[u S^+ v] =_{\text{def}} ([u] \rightarrow [v]) \wedge (X^+([u] \rightarrow [v]) W^+[u]).$$

The commitment of Formula 3 is fulfilled at a state s' , in the future of s , iff

$$\text{AF}^-(\text{Happ}(e_1) \wedge ((\text{lightning} \rightarrow X^+ \text{thunder}) \wedge (X^+(\text{lightning} \rightarrow X^+ \text{thunder}) W^+ \text{lightning})))$$

holds at s' . This formula becomes true at state s' if, and only if, for all paths starting from s and going through s' the first occurrence of a thunder after s is immediately followed by a lightning. Moreover, as it can easily be checked, the commitment of Formula 3 is violated at a state in the future of s as soon as it is the case that a lightning will be seen that is not immediately followed by a thunder.

These examples suggest that a satisfactory logic of commitment is induced by Definitions FC and VC, which specify the conditions under which a commitment is fulfilled or violated.

4. COMMUNICATIVE ACTS AND ACL MESSAGES

In the previous section we have defined the results of a number of commitment-manipulation actions, but we have not yet explained how these actions can be performed. The idea is the following: agents can perform commitment-manipulation actions by exchanging ACL messages, provided certain contextual conditions hold.

We consider as the fundamental unit of agent communication the *exchange of a message*. By this we mean that a message is sent by an agent, the *sender*, and received by another agent, the *receiver*. In turn, a message is viewed as a pair made up by a *type indicator* and a *body*. Type indicators (corresponding to KQML’s *performatives*) are constant symbols taken from a finite set, whose definition is part of the ACL specification. The body can be a sentence in some CL, whose abstract syntax is represented in our Semantic Language by a first-order term (see Section 3), or a more complex structure (for example a tuple of elements), typically including a CL sentence. When event e is an exchange of a message of type τ and body σ , sent by agent x to agent y , we write:

$$\text{Done}(e, x, \text{exch}(y, \tau, \sigma)).$$

Under given conditions, such an event implies a valid performance of a commitment-manipulation action. It is important to note that by associating commitment manipulation actions to messages, we formally specify a commitment-based semantics for an ACL. More precisely, the meaning of message $\langle \tau, \sigma \rangle$ is defined as the effect that exchanging $\langle \tau, \sigma \rangle$ has on the network of commitments binding the sender and the receiver. By defining a coherent set of message types in this way, it is possible to specify a Communicative Act Library with its associated semantics.

Below we analyze a few examples.

Informing

We assume that the body of an *inform* message is an arbitrary CL sentence. Informing is then defined as committing to the truth of the message body. More precisely, when agent x exchanges with agent y a message of type *inform* with an arbitrary CL sentence s as the body, agent x commits, relative to y , to the truth of s :

$$(\text{Inf}) \text{Done}(e, x, \text{exch}(y, \text{inform}, s)) \rightarrow \text{Done}(e, x, \text{mc}(x, y, s)).$$

Requesting

We assume that the body of a request message is an *action expression*, which describes the requested action by indicating its type, its actor, and possibly a temporal constraint. Concrete action expressions belonging to a specific CL should not be confused with the first-order term representing the abstract syntax of the expression in the Semantic Language. For example, here is an example of a possible concrete action expression describing the action type of actor *ag-1* moving object *obj-1* from location *loc-1* to location *loc-2* before *end-of-turn*:

```
(action :actor ag-1
        :type (move :object obj-1
                  :from loc-1
                  :to loc-2)
        :deadline end-of-turn)
```

The abstract syntax of this expression is given by the first-order term

$$u_1 = \text{before}(\text{done}(ag_1, \text{move}(obj_1, loc_1, loc_2)), \text{end-of-turn}),$$

which in turn can be translated into the Semantic Language formula

$$\lfloor u_1 \rfloor = \text{Done}(\neg, ag_1, \text{move}(obj_1, loc_1, loc_2)) \mathbf{B}^+ \text{end-of-turn},$$

where

$$\varphi \mathbf{B}^+ \psi =_{\text{def}} \neg(\neg\varphi \mathbf{U}^+ \psi).$$

With these assumptions, if term s represents the abstract syntax of an action expression, the semantics of a *request* message is defined by:

$$\text{(Req)} \quad \text{Done}(e, x, \text{exch}(y, \text{request}, s)) \rightarrow \text{Done}(e, x, \text{mp}(y, x, s)).$$

Accepting

We define *accepting* not only with respect to requests, but with respect to precommitments in general. We assume that the body of an acceptance message is a tuple including all the elements that uniquely identify the accepted precommitment:

$$\text{(Acc)} \quad \text{Done}(e', y, \text{exch}(x, \text{accept}, (e, y, x, s))) \wedge \text{Prec}(e, y, x, s) \\ \rightarrow \text{Done}(e', y, \text{ap}(e, y, x, s)).$$

Ordering

The difference between a request and an order is that while requests can be accepted or refused, orders cannot. In the terms of our approach, a request brings about a precommitment, and an order directly generates a commitment. To issue an order an agent must have powers that are not required to simply make a request; however, developing an articulated model of power relationships lies beyond the scope of this paper.

5. CONCLUSIONS AND FUTURE WORK

Even if social commitment has already been proposed [2,10] as a basis for the definition of ACL semantics, no full formal account of commitment has been put forward so far. In this paper we have presented a logical model of social commitment embedded in CTL^\pm , a logic of discrete time with no start or end points and branching in the future. The logical model of commitment has been completely specified at the level of formal semantics, and this has allowed us to prove some properties of commitment, expressed by valid formulae of our Semantic Language. Needless to say, we are just at the beginning of a long way. Below we point out some aspects that need to be further investigated.

Time

A sound and complete formal system for CTL^\pm has to be developed. This result should be easy to achieve by extending some known formal system for CTL^* . It would also be important to develop efficient model checking techniques for at least a sublanguage of CTL^\pm . Moreover, it may be worthwhile to consider an extension of CTL^\pm dealing with dense time, in order to give a more flexible account of interactions in a multiagent system.

Another important aspect is the expression of temporal qualifications in content sentences. Indeed, CTL^\pm is a powerful but very abstract temporal language. In many practical applications, like for example in the field of e-business, we can expect that temporal qualifications will be expressed with respect to some standard date system, like the Gregorian calendar. The critical point here is to specify a language by which common temporal qualifications can be represented in a natural and transparent way (see for example [8]).

Action

In this paper we have defined a minimal set of logical tools for the treatment of action. We feel, however, that it might be worthwhile to embed our logic of commitment in a richer language, possibly based on some version of dynamic logic.

An important point in our treatment is the association between an action and its results. In the case of commitment, this association has been represented by inserting an event-denoting term as the first argument of the *Comm* and *Prec* predicates. This solution has proved sufficient for our current goals, but may be difficult to extend to more complex situations.

Commitment

The main contribution of this paper is the logical treatment of commitment. Commitment is intrinsically a second-order concept, in that an agent commits to a proposition. Driven by a concern for simplicity, we decided to represent a commitment by a first-order predicate, and its content as a first-order term.

In designing our representation of commitment we have constantly kept in mind the reasons that motivate the development of a logical model in an area of Computer Science. In our opinion, the rigor and precision given by the use of logic is highly valuable, but should never bring us too far from practical applications, lest we give up the hope of influencing actual software practice.

We believe that our model of commitment can easily be translated into the conceptual toolkit and jargon of software designers. More precisely, commitments may be viewed as instances of a “commitment class,” whose instance variables contain: a reference to the commitment-inducing event (a message exchange), two references to agents (the debtor and the creditor), and an abstract representation of a CL sentence. In such a context, the commitment manipulation actions can be regarded as methods of the commitment class (see for example [6]), with formal specification given by Axioms MC, MP, and AP of Section 3.2. Continuing this line of thought, the definitions of fulfillment and violation can be viewed as the core specification of a “commitment management system,” which may be in charge of monitoring communicative exchanges in a multiagent system. Finally, the examples of Section 4 suggest that a Communicative Act Library may define a communicative act by specifying: (i), the general form of the class of messages by which the act is performed; (ii), relevant contextual conditions for a successful execution of the communicative act; and (iii), the effect of a successful execution of the communicative act, expressed in terms of commitment-manipulation actions. Developing a fully integrated logical and operational model of agent communication based on the notion of commitment is the main goal for our future research.

ACKNOWLEDGMENTS

We are grateful to Alessio Lomuscio and Marek Sergot for commenting on the first draft of this paper and for many interesting discussions, made possible by a British Council/CRUI grant for the year 2002.

REFERENCES

- [1] P. R. Cohen and H. J. Levesque. Rational interaction as the basis for communication. In P.R. Cohen, J. Morgan, and M. E. Pollack, editors, *Intentions in communication*, pages 221–256, MIT Press, Cambridge, MA, 1990.

- [2] M. Colombetti, A commitment-based approach to agent speech acts and conversation. In *Proc. Workshop on Agent Languages and Communication Policies, 4th International Conference on Autonomous Agents (Agents 2000)*, pages 21–29. Barcelona, Spain, 2000.
- [3] E. A. Emerson and J. Y. Halpern. ‘Sometimes’ and ‘Not Never’ revisited: On branching versus linear time temporal logic. *Journal of the ACM*, 33(1):151–178, 1986.
- [4] T. Finin, Y. Labrou, and J. Mayfield, KQML as an agent communication language. In J. Bradshaw, editor, *Software agents*. MIT Press, Cambridge, MA, 1995.
- [5] FIPA, Agent Communication Language. FIPA 2000 Specification, Foundation for Intelligent Physical Agents, www.fipa.org, 2000.
- [6] N. Fornara and M. Colombetti. Operational specification of a commitment-based communication language. In *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 02)*, Bologna, Italy, 2002.
- [7] Y. Labrou and T. Finin, Semantics and conversations for an agent communication language. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI’97)*, Nagoya, Japan, 1997.
- [8] H. J. Ohlbach and D. Gabbay. Calendar logic. *Journal of Applied Non-classical Logics*, 8(4):291–324, 1998.
- [9] H. Reichenbach, *Elements of Symbolic Logic*. MacMillan, New York, 1947.
- [10] M. P. Singh, Agent communication languages: Rethinking the principles. *IEEE Computer*, 31:40–47, 1998.
- [11] P. Yolum, M. P. Singh. Flexible protocol specification and execution: applying event calculus planning using commitments. In *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 02)*, Bologna, Italy, 2002.