An Integrated Semantics of Social Commitments and Associated Operations

M. El-Menshawy
Depart. of Electrical and Computer Engineering
Concordia University, Montreal, Canada
m_elme@encs.concordia.ca

J. Bentahar, R. Dssouli
Concordia Institute for Inf. Sys. Engineering
Concordia University, Montreal, Canada
{bentahar, dssouli}@ciise.concordia.ca

***Abstract*—In this paper, we develop a unified semantic model for social commitments and associated operations. We propose a logical model based on CTL^* with modalities of commitments and associated operations that represent the dynamic behavior of agents. Our semantics differs from the previous proposals in which the operations used to manipulate commitments (e.g. creation, fulfillment, violation, withdrawn, etc.) have always been defined as axioms or constrains on top of the commitment semantics. The advantage of this logical model is to gather the direct semantics of these operations and the semantics of social commitments (propositional and conditional) within the same framework. Furthermore, this paper proposes a new definition of assignment and delegation operations by looking at the content of the assigned and delegated commitment that could be different from the content of the original commitment in terms of deadline. Finally, to stress the soundness of the model, we prove that the proposed semantics satisfies some properties that are desirable when modeling commitment-based multiagent systems.**

I. Introduction

The importance of defining suitable and formal semantics of social commitments has been broadly recognized for multi-agent systems. Particulary, social commitments have been used for agent communication, coordination and artificial institutions [22]. In communication protocols, commitments can capture a high level meaning of interaction as opposed to low-level operational representation [24], [14], [9]. However, the meanings of messages exchanged between interacting agents can also be interpreted directly into social commitments in an operational semantics style [11], [6].

Some interesting semantic frameworks for social commitments have already been proposed using different approaches such as branching time logics (CTL^* and $CTL^{\pm}$) [2], [3], [13], [19], [25]. Recently, a model-theoretic semantics of social and dialogical commitments based on linear-time logic (LTL) has been introduced in [20] and the proposed postulates are reproduced in [7] to define semantics of commitment operations in distributed systems.

In general, two categories of semantic frameworks for social commitments can be distinguished. In the former category, commitment operations are formalized based on Singh's presentation [18] as axioms or constraints on top of commitment semantics [6], [9], [13], [14], [24]. These axioms are represented either as reasoning rules, updating rules or enforcing rules to evolve the truth of commitments' states. However, the real meanings of commitment operations themselves (e.g. *Create*, *Fulfill*, etc.) are not captured. In the later category, social commitments are formalized using object-oriented paradigm to advance the idea of commitments as data structure [11]. Thus, the main objective of defining clear, practical, and verifiable semantics of commitments and associated operations in the same framework is yet to be reached.

The objective of this paper aims to propose a new semantics not only for social commitments, but also for all the operations used to manipulate commitments. For verification issues and development methodologies of agent-oriented software engineering, the semantics of these operations should not be only captured by some enforced rules like in [13], but also integrated in the same framework [15]. In essence, this work is a continuation of our two previous publications [3], [15]. In the former one [3], we have developed a framework unifying commitments, actions on commitments and arguments that agents use to support their actions. In the second one [15], we have proposed a new logical semantics of social commitments and associated two-party operations based on Branching Space-Time (BST) logic. BST-logic provides this semantics with agent life cycle, space-like dimension and causal relation between interacting agents in the same (physical or virtual) space. Specifically, here we refine the semantics of some operations (e.g., *Create*, *Withdraw*, *Fulfill*) to overcome the *state explosion* problem that arises in [3], complete the life cycle of commitment operations introduced in [15], and define a new semantics of multi-party operations, such as *Delegate* and *Assign* using an extension of CTL^*.

The primary contribution of this paper is the unified logical model for commitments and associated operations. In fact, the paper introduces a new semantics of withdrawal, fulfilment, violation and release operations using the notions of accessible and non-accessible paths. New definitions of assignment and delegation operations are also proposed by taking into account the fact that the assigned and delegated commitment's deadline could be different form the deadline of the original commitment. Some desirable properties such as "the same social commitment (with the same identifier) cannot be created twice" or "if it is fulfilled, the commitment cannot be fulfilled again or withdrawn in the future" are captured, which makes the model sound. The proposed logical model uses a Kripke-like computational structure where accessibility relations for commitment modalities are defined using a computational interpretation, which makes our semantics computationally grounded [23] (this idea will be detailed later). This compotional interpretation is suitable for formal verification using model checking to verify interacting agent-based systems against given properties [4], but model checking algorithm is out of scope of this paper. Furthermore, the logical model presented here is expressive because the content of commitments are CTL*-like path formulae [10] and not state formulae and their semantics is expressed in terms of accessible paths and not in terms of deadlines.

The remainder of this paper is organized as follows. Section II describes the notion of social commitment and its formal notation extended from [15]. Given this context, Section III presents the syntax and semantics of the main elements of our logical model. Subsequently, Section IV proves some logical properties based on the defined semantics. Finally, the paper ends in Section V with a discussion of related work.

II. Social Commitments

A. Formal Notation

A commitment is an engagement made by one agent, the *debtor*, and directed towards another agent, the *creditor*, so that some fact is true. The debtor must respect and behave in accordance with his commitments. These commitments are contextual, manipulable and possibly conditional [18]. Furthermore, commitments are social and observable by all the participants. Consequently, social commitments (SC) are different from the agent's private mental states like beliefs, desires and intentions. Several approaches assume that agents will respect their commitments. However, this assumption is not always guaranteed in real-life scenarios (e.g in e-business) since violation can occur if agents are malicious, deceptive, malfunctioning or not reliable. Thus, it is natural to introduce violation operation of social commitments along with their satisfaction. We can also use a legal context of commitments to define rules that impose penalties on the debtors that violate their commitments. Below, we distinguish between two types of social commitments: propositional and conditional.

***Definition** 1: Propositional social commitments are related to the state of the world and denoted by $SC^p(id, Ag_1, Ag_2, \phi)$ where id is the commitment's identifier, Ag_1 is the debtor, Ag_2 is the creditor and ϕ is a well-formed formula (expressed in some logics) representing the commitment content.*

The basic idea is that Ag_1 is committed towards Ag_2 that the propositional formula ϕ is true. We suppose that the identifier id is unique so that there is at most one commitment with a particular identifier. In several situations, agents can only commit when some conditions are satisfied. Conditional commitments are introduced to capture this issue.

***Definition** 2: Conditional social commitments are denoted by $SC^c(id, Ag_1, Ag_2, \tau, \phi)$ where id, Ag_1, Ag_2, and ϕ have the same meanings as in Def.1 and τ is a well-formed formula representing the condition.*

B. Social Commitment Life Cycle

Having explained the formal definitions of commitments, in this section we present their life cycle to specify the relationship between commitments' states. Figure 1 describes this life cycle using UML state diagram. The life cycle proceeds as follows:

- The commitment could be *conditional* or *unconditional* (i.e. propositional). This is represented by the selection operator. The first operation an agent can perform on a commitment is creation. When created, a conditional commitment can move to the state of *created unconditional commitment* if the condition is true. Otherwise, the conditional commitment moves to the final state.
- When the unconditional commitment is created, then it may move to one of the following states: *fulfilled*, *violated*, *withdrawn*, *released*, *delegated* or *assigned*.
- The commitment can be *withdrawn* if the debtor decides to cancel it. Only the debtor is able to perform this action without any intervention from the creditor.
- The commitment is *fulfilled* if its content is satisfied by the debtor.
- The social commitment is *violated* if its content is violated by the debtor.
- The social commitment can be *released* by the creditor so that the debtor is no longer obliged to carry out his commitment.

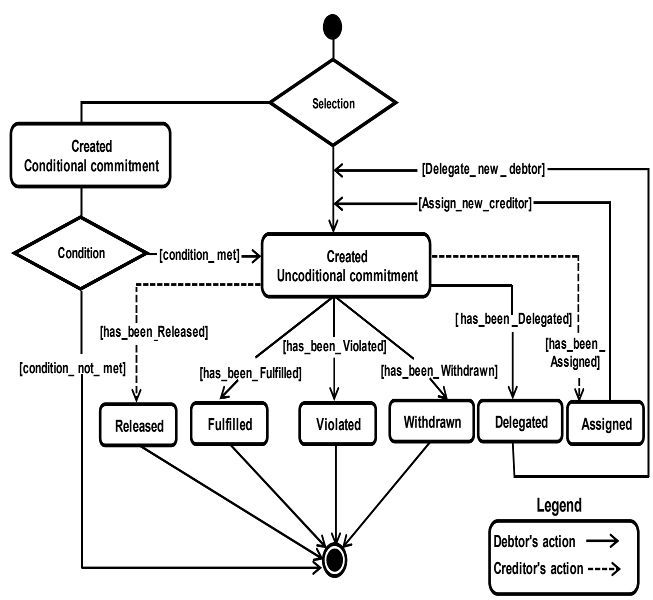

Fig. 1. Life cycle of social commitments

- The social commitment can be *assigned* by the creditor, which results in releasing this creditor from the commitment and having a new unconditional commitment with a new creditor.

- The social commitment can be *delegated* by the debtor, which results in withdrawing this debtor from the commitment and delegating his role to another debtor within a new commitment.

III. Logical Model of Social Commitments

This section introduces the syntax and semantics of the different elements of our formal language $\mathcal{L}$. This propositional language uses extended Computation Tree Logic (CTL*)[10] with past operators and two additional modalities SC^p for propositional and SC^c for conditional commitments, and Act for actions applied to commitments. We refer to the resulted branching time logic as CTL^{*sc}. The time in our model is discrete and branching in future to represent all choices that agents have when they participate in conversations and linear in the past. On the other hand, the dynamic behavior of agents is captured by actions these agents perform on different commitments during conversations.

A. Syntax

Let Φ_p be a set of atomic propositions and $\mathcal{ID}$ be a set of identifiers. $\mathcal{AGT}$ is a set of agent names and $\mathcal{ACT}$ is a set of actions used to manipulate commitments (e.g. *Create*, *Fulfill*, etc.). L and Act are nonterminals corresponding to $\mathcal{L}$ and $\mathcal{ACT}$, respectively. Furthermore, we use the following conventions: id, id_0, id_1, etc. are unique commitment identifiers in $\mathcal{ID}$, Ag_1, Ag_2, Ag_3, etc. are agent names in $\mathcal{AGT}$, p, q, etc. are atomic propositions in Φ_p and ϕ, ψ, etc. are formulae in $\mathcal{L}$. Table I gives the formal syntax of $\mathcal{L}$ expressed in Backus-Naur Form (BNF) grammar where "$\rightarrow$" and "$|$" are meta-symbols of this grammar.

L1. $L \rightarrow C \mid Act \mid p \mid \neg L \mid L \vee L \mid X^+L \mid X^-L \mid L\ U^+L \mid L\ U^-L \mid AL \mid EL$

L2. $C \rightarrow SC^p(id, Ag_1, Ag_2, L) \mid SC^c(id, Ag_1, Ag_2, L, L)$

L3. $Act \rightarrow Create(Ag, C) \mid Fulfill(Ag, C) \mid Violate(Ag, C) \mid Withdraw(Ag, C) \mid Release(Ag, C) \mid Assign(Ag_1, Ag_2, C) \mid Delegate(Ag_1, Ag_2, C)$

TABLE I
The Syntax of CTL^{*sc} Logic

The intuitive meanings of the most constructs are straightforward (from CTL* with next (X^+), previous (X^-), until (U^+), and since (U^-) operators). A and E are the universal and existential path-quantifiers over the set of all paths starting from the current moment. $A\phi$ (resp. $E\phi$) means that ϕ holds along all (some) paths starting at the current moment. Furthermore, there are some useful abbreviations based on temporal operators (X^+, X^-, U^+, U^-): (sometimes in the future) $F^+\phi \triangleq true\ U^+\phi$; (sometimes in the past) $F^-\phi \triangleq true\ U^-\phi$; (globally in the future) $G^+\phi \triangleq \neg F^+\neg\phi$ and (globally in the past) $G^-\phi \triangleq \neg F^-\neg\phi$. We also introduce $\mathcal{L}^- \subset \mathcal{L}$ as the subset of all formulae without temporal operators.

B. Semantics of CTL^{*sc}

1) The formal model:

Our model M for $\mathcal{L}$ is based on a Kripke-like structure with seven-tuple, $M = \langle \mathbb{S}, T, R, \mathbb{V}, \mathbb{R}_{scp}, \mathbb{R}_{scc}, \mathbb{F} \rangle$, where:

- $\mathbb{S} = \{s_0, s_1, s_2, \ldots\}$ is a set of states.
- $T : \mathbb{S} \rightarrow \mathcal{TP}$ is a function assigning to any state the corresponding time stamp where $\mathcal{TP}$ is a set of time points.
- $R \subseteq \mathbb{S} \times \mathbb{S}$ is a total transition relation, that is, $\forall s_i \in \mathbb{S},\ \exists\ s_j \in \mathbb{S} : (s_i, s_j) \in R$, indicating branching time. If there exists a transition $(s_i, s_j) \in R$, then we have $T(s_i) < T(s_j)$. A path P is an infinite sequence of states $P = \langle s_0, s_1, s_2, \ldots \rangle$ where $\forall\ i \in \mathbf{N},\ (s_i, s_{i+1}) \in R$. We denote the set of all paths by σ and the set of all paths starting at s_i by σ^{s_i}.
- $\mathbb{V} : \Phi_p \rightarrow 2^{\mathbb{S}}$ is a valuation function that assigns to each atomic proposition a set of states where the proposition is true.
- $\mathbb{R}_{scp} : \mathbb{S} \times \mathcal{AGT} \times \mathcal{AGT} \rightarrow 2^{\sigma}$ is a function producing an accessibility modal relation for propositional commitments.
- $\mathbb{R}_{scc} : \mathbb{S} \times \mathcal{AGT} \times \mathcal{AGT} \rightarrow 2^{\sigma}$ is a function producing an accessibility modal relation for conditional commitments.
- $\mathbb{F} : \mathcal{L} \rightarrow \mathcal{L}^-$ is a function associating to each formula in $\mathcal{L}$ a corresponding formula in $\mathcal{L}^-$.

The function $\mathbb{R}_{scp}$ associates to a state s_i the set of paths starting at s_i along which an agent commits towards another agent. These paths are conceived as merely "possible", and as paths where the commitments' contents made in s_i are true. The computational interpretation of this accessibility relation is as follows: the paths over the model M are seen as computations, and the accessible paths from a state s_i are the computations satisfying (i.e. computing) the formulae representing the contents of the commitments made at that state by a given agent towards another given agent. For example, if we have: $P' \in \mathbb{R}_{scp}(s_i, Ag_1, Ag_2)$, then the commitments that are made in the state s_i by Ag_1 towards Ag_2 are satisfied along the path $P' \in \sigma^{s_i}$.

$\mathbb{R}_{scc}$ is similar to $\mathbb{R}_{scp}$ and it gives us the paths along which the resulting unconditional commitment is satisfied if the underlying condition is true. Because it is possible to decide if a path satisfies a formula (see the semantics in this section), the model presented here is computationally grounded [23]. In fact, the accessible relations map commitment content formulae into a set of paths that simulate the behavior of interacting agents. The logic of propositional and conditional commitments is KD4 modal logic and the accessibility modal relations $\mathbb{R}_{scp}$ and $\mathbb{R}_{scc}$ are serials [3]. The function $\mathbb{F}$ is used to remove the temporal operators from a formula in $\mathcal{L}$. For example: $\mathbb{F}(X^+X^+p) = p$ and $\mathbb{F}(SC^p(id, Ag_1, Ag_2, X^+q)) = SC^p(id, Ag_1, Ag_2, q)$.

2) Semantics of social commitments:

Having explained our formal model, in this section we define the semantics of the elements of $\mathcal{L}$ relative to a model M, state s_i, and path P. The notation $\langle s_i, P\rangle$ refers to the path P starting at s_i meaning that $P \in \sigma^{s_i}$ where $P = \langle s_i, s_{i+1}, s_{i+2}, \ldots\rangle$. If P is a path starting at a given state s_i, then *prefix* of P starting at a state s_j $(T(s_j) < T(s_i))$ is a path denoted by $P \downarrow s_j$ and *suffix* of P starting at a state s_k $(T(s_i) < T(s_k))$ is a path denoted by $P \uparrow s_k$. Because the past is linear, s_j is simply a state in the unique past of s_i such that P is a part of $P \downarrow s_j$. s_k is in the future of s_i over the path P such that $P \uparrow s_k$ is part of P. If s_i is a state, then we assume that s_{i-1} is the previous state in the linear past $((s_{i-1}, s_i) \in R)$ and s_{i+1} is the next state on a given path $((s_i, s_{i+1}) \in R)$.

In the metalanguage, we use the following symbols: & means "and", $\Leftrightarrow$ means "is equivalent to" and $\Rightarrow$ means "implies that". The logical equivalence is denoted $\equiv$. As in CTL^*, we have two types of formulae: state formulae evaluated over states and path formulae evaluated over paths [10]. $M, \langle s_i\rangle \models \phi$ means "the model M satisfies the state formula ϕ at s_i". $M, \langle s_i, P\rangle \models \phi$ means "the model M satisfies the path formula ϕ along the path P starting at s_i". A state formula ϕ is satisfiable iff there are some M and s_i such that $M, \langle s_i\rangle \models \phi$. A path formula ϕ is satisfiable iff there are some M, P, and s_i such that $M, \langle s_i, P\rangle \models \phi$. A state formula is valid when it is satisfied in all models M, in all states s_i in M. A path formula is valid when it is satisfied in all models M, in all paths P in M, in all states s_i. The formal semantics of CTL^* and SC^p, SC^c modalities of our model is illustrated in Table II.

M1. $M, \langle s_i\rangle \models p$ iff $s_i \in \mathbb{V}(p)$ where $p \in \Phi_p$
M2. $M, \langle s_i\rangle \models \neg\, \phi$ iff $M, \langle s_i\rangle \not\models \phi$
M3. $M, \langle s_i\rangle \models \phi \vee \psi$ iff $M, \langle s_i\rangle \models \phi$ or $M, \langle s_i\rangle \models \psi$
M4. $M, \langle s_i\rangle \models A\phi$ iff $\forall\ P \in \sigma^{s_i} : M, \langle s_i, P\rangle \models \phi$
M5. $M, \langle s_i\rangle \models E\phi$ iff $\exists\ P \in \sigma^{s_i} : M, \langle s_i, P\rangle \models \phi$
M6. $M, \langle s_i\rangle \models SC^p(id, Ag_1, Ag_2, \phi)$ iff $\forall\ P \in \mathbb{R}_{scp}(s_i, Ag_1, Ag_2)\ M, \langle s_i, P\rangle \models \phi$
M7. $M, \langle s_i\rangle \models SC^c(id, Ag_1, Ag_2, \tau, \phi)$ iff $\forall\ P \in \mathbb{R}_{scc}(s_i, Ag_1, Ag_2)\ M, \langle s_i, P\rangle \models \tau \Rightarrow M, \langle s_i, P\rangle \models SC^p(id, Ag_1, Ag_2, \phi)$
M8. $M, \langle s_i, P\rangle \models \phi$ iff $M, \langle s_i\rangle \models \phi$
M9. $M, \langle s_i, P\rangle \models \neg\, \phi$ iff $M, \langle s_i, P\rangle \not\models \phi$
M10. $M, \langle s_i, P\rangle \models \phi \vee \psi$ iff $M, \langle s_i, P\rangle \models \phi$ or $M, \langle s_i, P\rangle \models \psi$
M11. $M, \langle s_i, P\rangle \models X^+\phi$ iff $M, \langle s_{i+1}, P \uparrow s_{i+1}\rangle \models \phi$
M12. $M, \langle s_i, P\rangle \models \phi\ U^+\ \psi$ iff $\exists\ j \geq i : M, \langle s_j, P \uparrow s_j\rangle \models \psi$ & $\forall\ i \leq k < j\ M, \langle s_k, P \uparrow s_k\rangle \models \phi$
M13. $M, \langle s_i, P\rangle \models X^-\phi$ iff $M, \langle s_{i-1}, P \downarrow s_{i-1}\rangle \models \phi$
M14. $M, \langle s_i, P\rangle \models \phi\ U^-\ \psi$ iff $\exists\ j \leq i : M, \langle s_j, P \downarrow s_j\rangle \models \psi$ & $\forall\ j < k \leq i\ M, \langle s_k, P \downarrow s_k\rangle \models \phi$

TABLE II
SEMANTICS OF CTL^* AND SC^p, SC^c MODALITIES

The semantics of state formulae is given from $M1$ to $M7$ and that of path formulae is given from $M8$ to $M14$. For space limit reasons, we only explain the semantics of formulae that are not in CTL*. $M6$ gives the semantics of propositional commitment operator, where the state formula is satisfied in the model M at s_i iff the content ϕ is true in all accessible paths P starting at s_i. Similarly, $M7$ gives the semantics of conditional commitment operator, where the state formula is satisfied in the model M at s_i iff in all accessible paths P if the condition τ is true, then the underlying unconditional commitment is also true. The semantics of past operators X^- and U^- is given by considering the linear past of the current state s_i as prefix of the path P.

3) Semantics of actions on social commitments: Having defined the semantics of commitments, below we define the semantics of operations used to manipulate those commitments and support flexibility. These operations are of two categorizes: two-party operations: *Create*, *Withdraw*, *Fulfill*, *Violate* and *Release*, and three-party operations: *Assign* and *Delegate* because *Assign* and *Delegate* need a third agent to which the new commitment is assigned or delegated. The context and detailed exposition of these operations are given in [13], [14], [18]. To simplify the notations used in the semantics, we suppose that these actions are momentary and do not need time to be performed. Technically, this can be represented by allowing actions to be performed on states or by supposing that transitions labeled by these actions are connecting two states s_i and s_j having the same time stamp $(T(s_j) = T(s_i))$. The first possibility is adopted in this paper. Although actions are momentary, action formulae are evaluated over paths. This is compatible with the philosophical interpretation of actions, according to which, by performing an action the agent selects a path or history among the available paths or histories at the moment of performing the action.

Creation action: the semantics of creation action of a propositional commitment (see Table III) is satisfied in the model M at state s_i along path P iff (i) the commitment is established in s_i (as a result of performing the momentary creation action); and (ii) the created commitment was not established in the past.

M15. $M, \langle s_i, P\rangle \models Create(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))$ iff
(i) $M, \langle s_i, P\rangle \models SC^p(id, Ag_1, Ag_2, \phi)$ &
(ii) $\forall\ j < i,\ M, \langle s_j\rangle \models \neg SC^p(id, Ag_1, Ag_2, \phi)$

TABLE III
SEMANTICS OF CREATION ACTION RELATIVE TO SC^p

The semantics of creation action of a conditional commitment (see Table IV) is defined in the same way.

M16. $M, \langle s_i, P\rangle \models Create(Ag_1, SC^c(id, Ag_1, Ag_2, \tau, \phi))$ iff
(i) $M, \langle s_i, P\rangle \models SC^c(id, Ag_1, Ag_2, \tau, \phi)$ &
(ii) $\forall\ j < i,\ M, \langle s_j\rangle \models \neg SC^c(id, Ag_1, Ag_2, \tau, \phi)$

TABLE IV
SEMANTICS OF CREATION ACTION RELATIVE TO SC^c

***Example** 1:* Let us consider a simple and modified case of NetBill protocol to illustrate the semantics of different action formulae. A customer (Cus) requests a quote for some goods (rfq), followed by the merchant (Mer) sending the quote as an offer. If the customer pays for goods, then the merchant will deliver the goods, withdraw (within a specified time), or not deliver. The customer can also release after receiving the quote (see Fig.2).

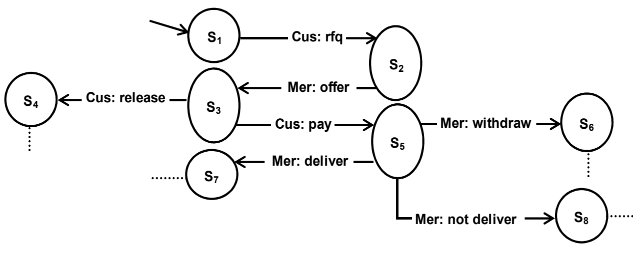

Fig. 2. Representation of NetBill

The offer message at state s_2 means that Mer creates a conditional commitment $Create(Mer, SC^c(id, Mer, Cus, pay, delivergoods))$ meaning that if the payment is received, then Mer commits to deliver the goods to Cus. $\langle s_2, s_3, s_4, \ldots \rangle$, $\langle s_2, s_3, s_5, s_6, \ldots \rangle$ and $\langle s_2, s_3, s_5, s_8, \ldots \rangle$ are not accessible paths for this commitment (i.e. are not in $\mathbb{R}_{scc}(s_2, Mer, Cus)$. However, $\langle s_2, s_3, s_5, s_7, \ldots \rangle$ is an accessible path (i.e. is in $\mathbb{R}_{scc}(s_2, Mer, Cus)$. As the condition is true through $\langle s_2, s_3, s_5, s_7, \ldots \rangle$ (the customer pays), the conditional commitment becomes unconditional: $SC^p(id, Mer, Cus, delivergoods)$ along the same accessible path. Before creating this unconditional commitment, Mer checks that it has not been created before, as there is no reason to create the same commitment twice.

Withdrawal action: the semantics of withdrawal action of a propositional commitment (see Table V) is satisfied in the model M at s_i along path P iff (i) the commitment was created in the past at s_j through the prefix $P \downarrow s_j$; (ii) this prefix is not one of the accessible paths via $\mathbb{R}_{scp}$; and (iii) at the current state s_i, there is still a possibility of satisfying the commitment since there is a path P' whose the prefix $P' \downarrow s_j$ is still accessible using $\mathbb{R}_{scp}$. Note that the first argument of $\mathbb{R}_{scp}$ is s_j where the commitment has been created. This is because the accessible paths start from the state where the commitment is created. Intuitively, when a commitment is withdrawn along a path, the prefix of this path from the creation state does not correspond to an accessible path (condition ii).

M17. $M, \langle s_i, P \rangle \models Withdraw(Ag_1, SC^p(id_1, Ag_1, Ag_2, \phi))$ iff
(i) $\exists j < i : M, \langle s_j, P \downarrow s_j \rangle \models Create(Ag_1, SC^p(id_1, Ag_1, Ag_2, \phi))$&
(ii) $P \downarrow s_j \notin \mathbb{R}_{scp}(s_j, Ag_1, Ag_2)$ &
(iii) $\exists P' \in \sigma^{s_i} : P' \downarrow s_j \in \mathbb{R}_{scp}(s_j, Ag_1, Ag_2)$

TABLE V
SEMANTICS OF WITHDRAWAL ACTION

Furthermore, a commitment can be withdrawn when its satisfaction is still possible (condition iii), which is captured by the existence, starting at the current moment, of an accessible path the agent can choose (see Fig.3). In other words, the agent Ag_1 has another choice at the current state, which is continuing in the direction of satisfying its commitment. We also note that withdrawing a commitment does not mean its content is false. For instance it can be accidentally true even if the current path is not amongst the accessible ones.

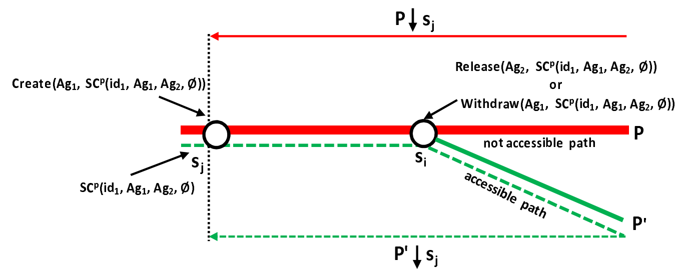

Fig. 3. Withdraw and Release actions at the state s_i along the path P

***Example** 2:* The merchant Mer, before delivering the goods to Cus, can withdraw the offer. Thus, there is no accessible path for the commitment between Mer and Cus at s_6. At the same time, Mer still has a possibility to satisfy its offer at state s_5 through the accessible paths $\langle s_2, s_3, s_5, s_7, \ldots \rangle$.

Fulfillment action: the semantics of fulfillment action (see Table VI) is defined in the same way as withdrawal. In (ii), the prefix $P \downarrow s_j$ of the current path P (along which the commitment has been created) is accessible using $\mathbb{R}_{scp}$; and in (iii), at the current state s_i, there is still a possible choice of not satisfying the commitment since a non-accessible path $P' \downarrow s_j$ exists. We notice that being accessible means that the content ϕ is true along $P \downarrow s_j$. As for withdrawal, fulfillment action makes sense only when a non-fulfilment action is still possible.

M18. $M, \langle s_i, P \rangle \models Fulfill(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))$ iff
(i) $\exists\, j < i : M, \langle s_j, P \downarrow s_j \rangle \models Create(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))$ &
(ii) $P \downarrow s_j \in \mathbb{R}_{scp}(s_j, Ag_1, Ag_2)$ &
(iii) $\exists\, P' \in \sigma^{s_i} : P' \downarrow s_j \notin \mathbb{R}_{scp}(s_j, Ag_1, Ag_2)$

TABLE VI
SEMANTICS OF FULFILLMENT ACTION

***Example** 3:* When the customer Cus pays for the goods and the merchant Mer delivers the goods (within a specified time), the merchant satisfies his commitment through the accessible path $\langle s_2, s_3, s_5, s_7, \ldots \rangle$. At the moment of satisfying the commitment, the merchant has still a possibility of not satisfying it through the non-accessible path $\langle s_2, s_3, s_5, s_6, \ldots \rangle$.

Violation action: the semantics of violation action (see Table VII) is almost similar to the semantics of withdrawal. The main difference is related to the truth of the commitment's content, which is false in the case of violation (condition ii). The fact that ϕ is false implies that $P \downarrow s_j$ is not accessible, but the reverse is not always true as explained above. Here again, violation makes sense when a choice of satisfying the commitment is still possible at the current state (condition iii).

M19. $M, \langle s_i, P\rangle \models Violate(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))$ iff
$(i)\ \exists\ j < i : M, \langle s_j, P \downarrow s_j\rangle \models Create(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))$ &
$(ii)\ M, \langle s_j, P \downarrow s_j\rangle \models \neg\phi$ &
$(iii)\ \exists\ P' \in \sigma^{s_i} : P' \downarrow s_j \in \mathbb{R}_{scp}(s_j, Ag_1, Ag_2)$

TABLE VII
SEMANTICS OF VIOLATION ACTION

***Example** 4:* When Cus pays for the goods, but Mer does not deliver them within a specified time, then Mer violates his commitment. Through the path $\langle s_2, s_3, s_5, s_8, \dots\rangle$ the content $delivergoods$ is false.

Release action: the semantics of release action (see Table VIII) is similar to the semantics of withdrawal. The only difference is that the release action is performed by the creditor while withdrawal is performed by the debtor (see Fig. 3).

M20. $M, \langle s_i, P\rangle \models Release(Ag_2, SC^p(id_1, Ag_1, Ag_2, \phi))$ iff
$(i)\ \exists j < i : M, \langle s_j, P \downarrow s_j\rangle \models Create(Ag_1, SC^p(id_1, Ag_1, Ag_2, \phi))$&
$(ii)\ P \downarrow s_j \notin \mathbb{R}_{scp}(s_j, Ag_1, Ag_2)$ &
$(iii)\ \exists\ P' \in \sigma^{s_i} : P' \downarrow s_j \in \mathbb{R}_{scp}(s_j, Ag_1, Ag_2)$

TABLE VIII
SEMANTICS OF RELEASE ACTION

***Example** 5:* The customer Cus, before paying for the goods, can release the offer. Thus, no accessible path exists between Cus and Mer from s_4. However, an accessible path still exists from s_3.

Assignment action: the semantics of assignment action of a propositional commitment (see Table IX) is satisfied in the model M at s_i along path P iff (i) the creditor Ag_2 releases the current commitment at s_i through P; and (ii) a new commitment with the same debtor and a new creditor appears at s_i, so that the formula $SC^p(id_1, Ag_1, Ag_3, \phi')$ is true at $M, \langle s_i\rangle$.

M21. $M, \langle s_i, P\rangle \models Assign(Ag_2, Ag_3, SC^p(id_0, Ag_1, Ag_2, \phi))$ iff
$(i)\ M, \langle s_i, P\rangle \models Release(Ag_2, SC^p(id_0, Ag_1, Ag_2, \phi))$ &
$(ii)\ \exists\ j<i : M, \langle s_j\rangle \models SC^p(id_0, Ag_1, Ag_2, \phi)$ &
$M, \langle s_i\rangle \models SC^p(id_1, Ag_1, Ag_3, \phi')$ such that:
$(1)\ \forall\ P' \in \sigma^{s_i},\ M, \langle s_j, P' \downarrow s_j\rangle \models \phi \Leftrightarrow M, \langle s_i, P'\rangle \models \phi'$ &
$(2)\ \mathbb{F}(\phi) \equiv \mathbb{F}(\phi')$

TABLE IX
SEMANTICS OF ASSIGNMENT ACTION

The most important issue in this semantics is that the content ϕ' of the new commitment is not necessarily the same as for the assigned one (ϕ), but there is a logical relationship between them. This is because the second commitment appears after the previous one. Thus, we need to consider the temporal component specifying the deadline of the first commitment. The logical relationship between ϕ and ϕ' is as follows: (1) ϕ' is true at the current state s_i through a given path P' iff ϕ is true at s_j where the original commitment has been created through the prefix $P' \downarrow s_j$; and (2) the two contents are logically equivalent when the temporal operators are removed. We consider the current state s_i in (1) because the new content ϕ' should be true starting from the moment where the new commitment is established (see Fig.4).

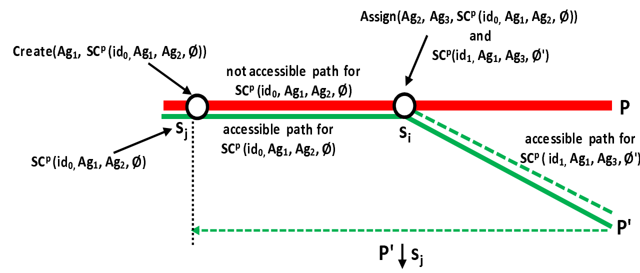

Fig. 4. Assign action at the state s_i along the path P

To clarify this issue, let us suppose that the content of the assigned commitment is $\phi = X^+X^+p$ where p is an atomic proposition and the assignment action takes place at the next moment after the creation action. The content of the resulting commitment should be then $\phi' = X^+p$, which is the content we obtain by satisfying the conditions (1) and (2). By (1) we have X^+p is true at a state s_i through a path P' iff X^+X^+p is true at the state s_j ($s_j = s_{i-1}$) through $P' \downarrow s_j$; and by (2) we have $\mathbb{F}(X^+X^+p) \equiv \mathbb{F}(X^+p)$. The second condition is added to guarantee that the relationship between the contents is not arbitrary.

***Example** 6:* Suppose Cus commits to pay \$200 to Mer in two days. After one day, Mer, for some reasons, assigns this commitment to Mer_1 (we suppose that there is an agreement between Cus and Mer_1 about this commitment). Thus, Mer releases the commitment with Cus and a new commitment between Cus and Mer_1 is established to pay the \$200 after only one day.

In the semantics proposed in previous frameworks (for example in [13], [14] and [20]), the two commitments have the same content, which implicitly suppose that the creation of the commitment and its assignment take place at the same moment. The previous example cannot be managed using this assumption.

Delegation action: the semantics of delegation action (see Table X) is similar to the semantics of assignment. The only difference is that delegation is performed by the debtor while assignment is performed by the creditor. Therefore, instead of release, the semantics is defined in terms of withdraw.

***Example** 7:* Suppose Cus commits to pay \$200 to Mer

in two days. After one day, Cus, for some reasons, delegates this commitment to a financial company ($Bank$) to pay the \$200 to Mer on his behalf. Thus, Cus withdraws his commitment and a new commitment between $Bank$ and Mer is established to pay the \$200 after only one day.

M22. $M, \langle s_i, P\rangle \models Delegate(Ag_1, Ag_3, SC^p(id_0, Ag_1, Ag_2, \phi))$ iff
(i) $M, \langle s_i, P\rangle \models Withdraw(Ag_1, SC^p(id_0, Ag_1, Ag_2, \phi))$ &
(ii) $\exists\, j < i : M, \langle s_j\rangle \models SC^p(id_0, Ag_1, Ag_2, \phi)$ &
$M, \langle s_i\rangle \models SC^p(id_1, Ag_3, Ag_2, \phi')$ such that:
(1) $\forall\, P' \in \sigma^{s_i},\ M, \langle s_j, P'{\downarrow}s_j\rangle \models \phi \Leftrightarrow M, \langle s_i, P'\rangle \models \phi'$ &
(2) $\mathbb{F}(\phi) \equiv \mathbb{F}(\phi')$

TABLE X
SEMANTICS OF DELEGATION ACTION

IV. COMMITMENTS' PROPERTIES

The aim of this section is to prove that the model aforementioned in the previous section presents a satisfactory "logic of commitment". We show some of desirable properties related to the semantics of actions on social commitments that are fundamental for soundness considerations where alignment [7] among interacting agents in distributed systems is satisfied. In the rest of this paper, the set of all models is denoted $\mathcal{M}$.

***Proposition** 1: If a commitment is created, then it has been never created before.*

$$AG^+\big[Create(Ag_1, SC^p(id, Ag_1, Ag_2, \phi)) \Rightarrow X^-G^-\neg Create(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))\big]$$

Proof: Let M be a model in $\mathcal{M}$, s_i be a state in $\mathbb{S}$, and P be a path in σ. Also, suppose that:
$M, \langle s_i, P\rangle \models Create(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))$
(Semantics of creation action)
$\Rightarrow \forall j < i\ M, \langle s_j, P\rangle \models \neg SC^p(id, Ag_1, Ag_2, \phi)$
(Semantic calculus)
$\Rightarrow M, \langle s_i, P\rangle \models X^-G^-\neg SC^p(id, Ag_1, Ag_2, \phi)$ **(1)**
Let us now suppose that:
$M, \langle s_i, P\rangle \models X^-F^-\ Create(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))$
(Semantics of creation action)
$\Rightarrow M, \langle s_i, P\rangle \models X^-F^-\ SC^p(id, Ag_1, Ag_2, \phi)$
There is then contradiction with (1). Consequently:
$M, \langle s_i, P\rangle \models X^-G^-\neg Create(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))$ ■

As a direct consequence of this proposition, we have the following lemma:

***Lemma** 1: Once created, a commitment cannot be created again in the future.*

$$AG^+\big[Create(Ag_1, SC^p(id, Ag_1, Ag_2, \phi)) \Rightarrow X^+AG^+\neg Create(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))\big]$$

Proof: Let us suppose that the negation is true. Therefore:
$EF^+\big[Create(Ag_1, SC^p(id, Ag_1, Ag_2, \phi)) \wedge \neg(X^+AG^+\neg Create(Ag_1, SC^p(id, Ag_1, Ag_2, \phi)))\big]$
(Semantic calculus)
$\Rightarrow EF^+\big[Create(Ag_1, SC^p(id, Ag_1, Ag_2, \phi)) \wedge X^+EF^+Create(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))\big]$

Consequently, there is a contradiction with Proposition 1 as the second creation cannot take place since there is a creation in its past. Thus, we are done. ■

***Proposition** 2: Once withdrawn, a commitment cannot be withdrawn again in the future.*

$$AG^+\big[Withdraw(Ag_1, SC^p(id, Ag_1, Ag_2, \phi)) \Rightarrow X^+AG^+\neg Withdraw(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))\big]$$

Proof: Let M be a model in $\mathcal{M}$, s_i be a state in $\mathbb{S}$, and P be a path in σ. Also, suppose that:
$M, \langle s_i, P\rangle \models Withdraw(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))$
(Semantics of withdrawal and creation actions)
$\Rightarrow\ \exists j < i : M, \langle s_j, P \downarrow s_j\rangle \models SC^p(id, Ag_1, Ag_2, \phi)$ &
$P \downarrow s_j \notin \mathbb{R}_{scp}(s_j, Ag_1, Ag_2)$ **(2)**
Let us now suppose that:
$M, \langle s_i, P\rangle \models X^+EF^+\ Withdraw(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))$
(Semantics of withdrawal action, X^+ and EF^+)
$\Rightarrow \exists k > i\ \&\ \exists P' \in \sigma^{s_k} : P'{\downarrow}s_j \in \mathbb{R}_{scp}(s_j, Ag_1, Ag_2)$
There is then contradiction with (2) because P' is a suffix of P. Consequently:
$M, \langle s_i, P\rangle \models \neg X^+EF^+Withdraw(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))$
$\Rightarrow M, \langle s_i, P\rangle \models X^+AG^+\neg Withdraw(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))$ ■

In the same way, we can prove the following three propositions (3,4 and 5) using the semantics of withdraw, fulfillment, violation and release actions, X^+ and EF^+.

***Proposition** 3: Once withdrawn, a commitment cannot be fulfilled in the future.*

$$AG^+\big[Withdraw(Ag_1, SC^p(id, Ag_1, Ag_2, \phi)) \Rightarrow X^+AG^+\neg Fulfill(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))\big]$$

***Proposition** 4: Once withdrawn, a commitment cannot be violated in the future.*

$$AG^+\big[Withdraw(Ag_1, SC^p(id, Ag_1, Ag_2, \phi)) \Rightarrow X^+AG^+\neg Violate(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))\big]$$

***Proposition** 5:* Once withdrawn, a commitment cannot be released in the future.

$$AG^+\big[Withdraw(Ag_1, SC^p(id, Ag_1, Ag_2, \phi)) \Rightarrow X^+AG^+\neg Release(Ag_2, SC^p(id, Ag_1, Ag_2, \phi))\big]$$

***Proposition** 6: Once withdrawn, a commitment cannot be assigned in the future.*

$$AG^+\big[Withdraw(Ag_1, SC^p(id, Ag_1, Ag_2, \phi)) \Rightarrow \forall Ag \in \mathcal{AGT}\ X^+AG^+\neg Assign(Ag_1, Ag, SC^p(id, Ag_1, Ag_2, \phi))\big]$$

Proof: Let M be a model in $\mathcal{M}$, s_i be a state in $\mathbb{S}$, and P be a path in σ. Also, suppose that:
$M, \langle s_i, P\rangle \models Withdraw(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))$
(Proposition 5)
$\Rightarrow M, \langle s_i, P\rangle \models X^+AG^+\neg Release(Ag_2, SC^p(id, Ag_1, Ag_2, \phi))$
(Semantics of assignment action)

$\Rightarrow \forall Ag \in \mathcal{AGT}$
$M, \langle s_i, P\rangle \models X^+ AG^+ \neg Assign(Ag_2, Ag, SC^p(id, Ag_1, Ag_2, \phi))$ ■

Using Proposition 2 and the semantics of delegation action, we can prove the following proposition:

Proposition *7: Once withdrawn, a commitment cannot be delegated in the future.*

$$AG^+\big[Withdraw(Ag_1, SC^p(id, Ag_1, Ag_2, \phi)) \Rightarrow \forall Ag \in \mathcal{AGT}$$
$$X^+ AG^+ \neg Delegate(Ag_1, Ag, SC^p(id, Ag_1, Ag_2, \phi))\big]$$

As for previous propositions for withdrawal action, we can also prove the following proposition for fulfillment action:

Proposition *8: Once fulfilled, a commitment cannot be fulfilled again, withdrawn, violated, released, assigned or delegated in the future.*

$$\begin{aligned}AG^+\big[&Fulfill(Ag_1, SC^p(id, Ag_1, Ag_2, \phi)) \Rightarrow \forall Ag \in \mathcal{AGT}\\ &X^+ AG^+ \neg \big[Fulfill(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))\\ &\quad \vee Withdraw(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))\\ &\quad \vee Violate(Ag_1, SC^p(id, Ag_1, Ag_2, \phi))\\ &\quad \vee Release(Ag_2, SC^p(id, Ag_1, Ag_2, \phi))\\ &\quad \vee Assign(Ag_2, Ag, SC^p(id, Ag_1, Ag_2, \phi))\\ &\quad \vee Delegate(Ag_1, Ag, SC^p(id, Ag_1, Ag_2, \phi))\big]\big]\end{aligned}$$

V. Discussion and Related Work

A. Discussion

Our logical model is useful when developing agent communication languages (ACL) thanks to its foundation based on social semantics. Unlike mentalistic semantics that specifies the semantics of communicative acts in terms of pre- and post-conditions contingent on so-called agent's mental states (e.g. beliefs, desires and intentions), this social semantics can be verified [4], [21], [22]. This is because our semantics allows for tracing the status of existing commitments at any point in time given observed actions. In commitment protocols, social commitments capture a high meaning of interactions and provide a useful level of abstraction. In this sense, Yolum and Singh [24] have used commitment operations to show how to build and execute commitment protocols and how to reason about them using event calculus. In the same way, Mallya and Singh [14] have showed how to reason about subsumption among commitment protocols and how to refine and aggregate protocols based on commitment semantics and operations. Also, Desai and Singh [9] have studied a composition of commitment protocols and concurrent operations. Our proposal belongs to the same line of research and can be used to specify commitment protocols in terms of creation and manipulation commitments using accessibility relation and the principle properties of commitment.

In fact, our framework provides a unified semantics of social commitments and associated operations that can be used to enhance, e.g., Tropos methodology [16], where Tropos is an agent-oriented software methodology, via capturing the meaning of interactions in terms of task dependencies among communicating agents. Also, in [20] Singh has delineated the model-theoretic semantics of commitments by postulating some rules as ways of using and reasoning with commitments. This model combines two commitments (practical and dialogical), in the sense that when a commitment arises within an argument and the content is satisfied with the same argument, then practical commitment would be satisfied. However, this model does not include the semantics of commitment operations. Chopra and Singh [7] have used the theoretical model proposed in [20] to study the semantics of commitment operations with message patterns that implement commitment operations with some constraints on agents' behaviors to tackle the problem of autonomy in distributed systems. This semantics is expressed in terms of the set of propositions that can be inferred from the observation sequence that agents sent or received. Moreover, this semantics must correspond to the postulates introduced in [20]. However, the formal language of those postulates is based on enhancing LTL (linear-time logic) with two commitment modalities. Thus, this language is less expressive than the formal language introduced here, which is more compatible with agent choices. Furthermore, our semantics is based on Kripke structure with accessibility relations, which enables us to prove that the proposed model is computationally grounded [23] and to verify this semantics. Finally, in [5], the propositional commitment is fulfilled when the creditor does not believe that the commitment's content is false and he cannot challenge it anymore. However, this semantics uses mental states which cannot be verified. The semantics defined here for conditional commitments is different from the semantics defined in [17] and [20]. In [17], conditional commitments are considered as intentions, while commitments as social notion are different from private intentions. In [20], Singh models conditional commitments as fundamental and unconditional commitments as special cases where the antecedent is true. In our semantics, the conditional commitments are transformed into propositional commitments in all accessible paths where the underlying condition is true.

B. Related Work

Let us now focus on comparing the proposed semantics for commitment operations with the related ones. The semantics proposed here is close to the semantics introduced in [3], but does not suffer from the "recursion" problem, which is the main problem in [3]. Recursion means the semantics of one operation depends on the semantics of one or more other operations. For example, in [3], a propositional commitment is satisfied along a path P at a state s_i iff it is active in this state along this path, and it was already created at a state s_j, and along this path from the state s_j the commitment content is true. Also, a commitment is active iff this commitment was already created and until the current moment the commitment was not withdrawn. Consequently, the model checking technique for this logic is very complex and probably suffers from the *state explosion* problem in the early phases. On the contrary, the semantics we presented here is independent, for each operation, of the semantics of other operations. Furthermore, our semantics of commitment operations is different from the ones given in [6], [8], [13] and [14]. Particularly (as discussed in Section III-B3), assignment and delegation operations should consider that the content of the new resulting commitment could be different from, and has a logical relationship with the content of the assigned and delegated commitment. This issue is not captured in previous frameworks. In addition, unlike our semantics, the violation operation has been disregarded.

C. Directions: Theoretical and Practical

Here, we outline two promising directions of future work.

Theoretically. We plan to improve our proposed semantics by removing the simplification based on the supposition that actions are only momentary and considering time frames between the execution of actions. We also plan to study the commitment operations needed to handle meta-commitments, that is commitments about commitments, that often arise in real-life scenarios. The proposed semantics would be augmented with argumentation to enhance the Tropos methodology, which we plan to apply for modeling and establishing communities of web services introduced in [12].

Practically. We intend to integrate our logical model with the logic of agent programs developed in [1] for the implementation of agents. Furthermore, a rigorous semantics opens up the way for improving the verification of logic-based protocols that govern a set of autonomous interacting agents against given properties. The mainstream step in this regard would be to map the commitment semantics introduced here to conventional verification technologies. Our semantics is based on Kripke structures (like interpreted systems). Currently model checking techniques work best for logics whose semantics is given via accessibility relations with the extension of CTL^* as proposed in [4]. Two complementary software tools are suggested to implement model checking algorithms to verify whether or not the model M satisfies the proposed commitments' properties (i.e., $M \models \phi$). Model checking algorithm for our logic can be implemented via $NuSMV$ tool, which is the best-known for CTL^*. On the other hand, the proposed logic and associated properties, which need to be checked, can be specified as tableau-based rules. Such rules provide a simple decision procedure for the logic and overcome model checking algorithm from *state explosion* problem. As proposed in [4], the verification method could be based on the translation of formulae into a variant of alternating tree automata called Büchi tableau automata (ABTA).

ACKNOWLEDGEMENTS

We would like to thank NSERC (Canada), NATEQ and FQRSC (Québec) for their financial support.

REFERENCES

[1] N. Alechina, M. Dastani, B.S. Logan and J.-J. Ch. Meyer. A Logic of Agent Programs. In Proc. of the Twenty-Second AAAI Conf. on Artificial Intelligence (AAAI), pp.795-800, 2007.

[2] J. Bentahar, B. Moulin, J.-J. Ch. Meyer and B. Chaib-draa. A Logical Model for Commitment and Argument Network for Agent Communication. In Proc. of the Int. Joint Conf. on AAMAS, pp.792-799, 2004.

[3] J. Bentahar, B. Moulin, J.-J. Ch. Meyer and Y. Lespérance. A New Logical Semantics for Agent Communication. K. Inoue, K. Satoh, F. Toni (eds.), Computational Logic in Multi-Agent Systems. LNAI 4371, pp.151-170, 2007.

[4] J. Bentahar, J.-J. Ch. Meyer and W. Wan. Model Checking Communicative Agent-based Systems. In Knowledge-Based Systems, Special Issue on Intelligent Software Design, vol.22(3), Elsevier, pp.142-159, 2008.

[5] G. Boella, R. Damiano, J. Hulstijn and L. Torre. Distinguishing Propositional and Action Commitment in Agent Communication. In Proc. of the Workshop on Comput. Modles of Natural Argument (CMNA'07).

[6] A.K. Chopra and M.P. Singh. Constitutive Interoperability. In Proc. of the Int. Joint Conf. on AAMAS, pp.797-804, 2008.

[7] A.K. Chopra and M.P. Singh. Multiagent Commitment Alignment. In Proc. of the 8th Int. Joint Conf. on Autonomous Agents and MultiAgent Sys. (AAMAS). May 2009.

[8] A.K. Chopra and M.P. Singh. Nonmonotonic Commitment Machines. In Dignum F. (eds) Advances in Agent Communication. LNAI 2922, pp.183-200, 2004.

[9] N. Desai, A.K. Chopra and M.P. Singh. Representing and Reasoning About Commitments in Business Processes. In Proc. of the 22nd Conf. AAAI, pp.1328-1333, 2007.

[10] E.A. Emerson and J.Y. Halpern. Sometimes and not never, Revisited: on Branching versus Linear time Temporal Logic. Journal ACM (JACM) vol.33(1), pp.151178, 1986.

[11] N. Fornara and M. Colombetti. Operational Specification of a Commitment-based Agent Communication Language. In Proc. of the Int. Joint Conf. on AAMAS, pp.535-542, 2002.

[12] Z. Maamar, S. Subramanian, J. Bentahar, P. Thiran P and D. Benslimane. An Approach to Engineer Communities of Web Services Concepts, Architecture, Operation and Deployment. In the Int. Journal of E-Business Research, vol.5(4), IGI Global.

[13] A.U. Mallya, P. Yolum and M.P. Singh. Resolving Commitments among Autonomous Agents. F. Dignum (eds.), Advances in Agent Communication. LNAI 2922, pp.166-182, 2004.

[14] A.U. Mallya and M.P. Singh. An Algebra for Commitment Protocols. Autonomous Agents and Multi-Agent Systems, vol.14(2), pp.143-163, 2007.

[15] M. El-Menshawy, J. Bentahar and R. Dssouli. A New Semantics of Social Commitments using Branching Space-Time Logic. In Proc. of the IEEE/WIC/ACM Inter. Conf. on Inte. Agent Technology (IAT'09), Logics for Inte. Agents and Multi-Agent Systems, Milano, Italy, September 15-18, 2009.

[16] P. Bresciani, A. Perini, P. Giorgini, F. Giunchiglia and J. Mylopoulos. Tropos: An Agent-Oriented Software Development Methodology. Autonomous Agents and Multi-Agent Systems, vol.8(3), pp.203-236, 2004.

[17] M.K. Shakil and L. Yves. On the Semantics of Conditional Commitment. In Proc. of the Int. joint Conf. on AAMAS, pp.1337-1344, 2006.

[18] M.P. Singh. An Ontology for Commitments in Multiagent Systems: toward a Unification of Normative Concepts. AI and Law, vol.7, pp.97113, 1999.

[19] M.P. Singh. A Social Semantics for Agent Communication Languages. Issues in Agent Communication, Dignum, F. and Greaves, M. (eds.), LNAI 1916, pp.31-45, 2000.

[20] M.P. Singh. Semantical Considerations on Dialectical and Pratical Commitments. In Proc. of the 23rd Conf. AAAI, pp. 176-181, 2008.

[21] M.P. Singh. Agent Communication Languages: Rethinking the Principles. IEEE Computer, vol.31(12), pp.40-47, 1998.

[22] P. Torroni, F. Chesani, P. Yolum, M. Gavanelli, M.P. Singh, E. Lamma, M. Alberti and P. Mello. Modelling Interactions via Commitments and Expectations. In Handbook of Research on Multi-Agent Systems: Semantics and Dynamics of Organizational Models, ch.11, V. Dignum (eds.), pp.263-284, 2009.

[23] M. Wooldridge. Computationally Grounded Theories of Agency. In E. Durfee, editor, In Proc. of the 4th Int. Conf. on Multi-Agent Sys. (ICMAS 2000). IEEE Press, 2000.

[24] P. Yolum and M.P. Singh. Flexible Protocol Specification and Execution: Applying Event Calculus Planning using Commitments. In Proc. of the Int. Joint Conf. on AAMAS, pp.527-534, 2002.

[25] M. Verdicchio and M. Colombetti. A Logical Model of Social Commitment for Agent Communication. In Proc. of the Int. Joint Conf. on AAMAS, pp.528-535, 2003.