Temporalised Epistemic Logic for Reasoning about Agent-Based Systems

Author

Ma, Ji, Orgun, Mehmet A, Sattar, Abdul, Adi, Kamel

Published

2009

Conference Title

2009 IEEE/WIC/ACM INTERNATIONAL JOINT CONFERENCES ON WEB INTELLIGENCE (WI) AND INTELLIGENT AGENT TECHNOLOGIES (IAT), VOL 3

Link to published version

http://portal.acm.org/citation.cfm?id=1632213

Griffith Research Online

https://research-repository.griffith.edu.au

Temporalised Epistemic Logic for Reasoning about Agent-Based Systems

Ji Ma*, Mehmet A. Orgun†, Abdul Sattar‡ and Kamel Adi*

*Department of Computer Science and Engineering, Université du Québec en Outaouais, Québec, Canada
Email: {maxj01,kamel.adi}@uqo.ca
†Department of Computing, Macquarie University, NSW 2109, Australia
Email: mehmet@science.mq.edu.au
‡School of Information and Communication Technology, Griffith University, Queensland 4111, Australia
Email: A.Sattar@griffith.edu.au

Abstract—For modelling and verifying agent systems, many researchers have proposed different logical systems. Since agent-based systems are designed to operate in dynamic environments such as the Internet, it is also important to model the temporal aspects of such systems in a systematic way. In this paper, we use a temporalised epistemic logic called TEL for formalising agent-based systems. We also propose a labelled tableau system and a model checking method for this logic. With logic TEL and its associated proof system, we are able to reason about, and verify agent systems operating in dynamic environments.

Keywords– Multi-agent systems, verification, model checking, temporal epistemic logic, tableaux.

I. INTRODUCTION

It has been argued that any logical system used for modelling active agents should be a combined system of logics of knowledge, belief, time and norms [1] since these are among the essential concepts to be reasoned about. Combining epistemic logic with temporal dimension is a particular interesting issue, since there is a need to study how knowledge changes over time in many real life applications.

There have been several methods and techniques proposed for combining logics in a systematic way, such as fusion [2], fibring [1], products [3], temporalisation [4], and hierarchy combination [5]. Depending on the choice of the logics to be combined and the method of combination of logics, different logics with different expressive capabilities may result. Model checking is a method for automated verification of finite state systems. Bordini *et al* [6] discussed how to verify multi-agent programs by model checking using the SPIN model checker. Franceschet et al. [7] discussed how to develop a model checker for a combined logic. Layouni et al. [8] proposed a model checking method for conflict detection.

In this paper, we propose a temporal epistemic logic, by combining an epistemic logic and a linear temporal logic using the temporalising technique proposed by Finger and Gabbay [4]. Temporalisation is a methodology for adding a temporal logic on top of another to generate a new logic system with temporal features. The resulting temporal epistemic logic can be used to specify and reason about epistemic properties that change through time. We also present a labelled tableau proof system for the logic. With the temporal epistemic logic and its associated reasoning techniques and model checker,

we are able to show how to reason about certain properties of agent-based systems operating in dynamic environments.

The rest of this paper is organised as follows: Section II introduces the temporal epistemic logic. Section III proposes a labelled tableau system for the logic which can be used to verify the properties of agent-based systems. Section IV proposes a model checking algorithm for the logic which can be used for automated verification. Section V concludes the paper.

II. TEMPORALISED EPISTEMIC LOGIC

A temporal epistemic logic has two classes of modal operators: (1) knowledge operators, and (2) temporal operators. The knowledge operators are those of an epistemic logic, and the temporal operators are those of a temporal logic. Each agent has an associated epistemic operator, for instance, the operator, $\mathbf{K}_i$ is used for declaring the knowledge of agent i. We adopt the temporal logic with clocks proposed by Liu and Orgun [9]. The temporal logic with clocks contains two temporal operators: **first** and **next**, which refer to the initial moment and the next moment in time with respect to given clocks respectively. Since the temporalised epistemic logic is obtained by using temporalisation [4], which is a hierarchy combination technique for combining logics, the temporal operators **first** and **next** can never appear within the scope of a knowledge operator $\mathbf{K}_i$.

A time model for a temporal logic is defined as a tuple $\mathscr{M}_{TL} = \langle ck, <, \pi \rangle$, where $ck = \langle t_0, t_1, t_2, \dots \rangle$ is a clock, $<$ is the usual ordering relation over ck, and π is an assignment function that gives a value $\pi(t,p) \in \{true, false\}$ for any time point t in ck and any atomic formula p. The assignment function π interprets all the terms given in atomic formula p in a standard way. Below we give the semantics of the temporal operators of the temporal logic:

- $\mathscr{M}_{TL}, t_i \models \mathbf{first}\ \varphi$, iff $\mathscr{M}_{TL}, t_0 \models \phi$.
- $\mathscr{M}_{TL}, t_i \models \mathbf{next}\ \varphi$, iff $\mathscr{M}_{TL}, t_{i+1} \models \phi$.

A knowledge model is defined as a tuple $\mathscr{M}_{EL} = \langle S, R_1, \dots, R_n, \pi \rangle$, where S is the set of states or possible worlds; and each R_i, $i = 1, \dots, n$, is the possibility relation according to agent i. R_i is a non-empty set consisting of state pairs (s_i, s_j) such that $(s_i, s_j) \in R_i$ iff, at state s_i, agent i

considers the state s_j possible; and π is a assignment function, which gives a value $\pi(s,p) \in \{true, false\}$ for any $s \in S$ and atomic formula p. Again, the assignment function π interprets all the terms given in atomic formula p in a standard way. Formula ϕ is satisfiable in the model $\mathscr{M}_{EL}$ if there exists $s \in S$ such that $\mathscr{M}_{EL}, s \models \phi$. Below we give the semantics of the knowledge operators of the epistemic logic:

- $\mathscr{M}_{EL}, s_i \models \mathbf{K}_i \phi$, iff for all s_j where $(s_i, s_j) \in R_i$, $\mathscr{M}_{EL}, s_j \models \phi$.

For combining an epistemic logic and a temporal logic using temporalisation technique, let $\mathscr{L}_{EL}$ be the set of formulas of the epistemic logic, then $\mathscr{L}_{EL}$ is defined as follows:

1. If ϕ is an atomic formula of the epistemic logic, then $\phi \in \mathscr{L}_{EL}$.
2. If $\phi(X) \in \mathscr{L}_{EL}$ where $\phi(X)$ is a formula containing a free variable X, then $(\forall X)\phi(X) \in \mathscr{L}_{EL}$.
3. If $\phi \in \mathscr{L}_{EL}$, then $\mathbf{K}_i\ \phi \in \mathscr{L}_{EL}$, $(1 \leq i \leq n)$.

Let $\mathscr{L}_{TEL}$ be the set of formulas of the temporalised epistemic logic . Then $\mathscr{L}_{TEL}$ is defined as follows:

1. If $\phi \in \mathscr{L}_{EL}$, then $\phi \in \mathscr{L}_{TEL}$.
2. If $\phi \in \mathscr{L}_{TEL}$, then $\neg\phi \in \mathscr{L}_{TEL}$.
3. If $\phi, \psi \in \mathscr{L}_{TEL}$, then $\phi \wedge \psi \in \mathscr{L}_{TEL}$.
4. If $\phi \in \mathscr{L}_{TEL}$, then $\mathbf{first}\ \phi \in \mathscr{L}_{TEL}$, and $\mathbf{next}\ \phi \in \mathscr{L}_{TEL}$.

Let $\mathscr{K}^+$ be a class of Kripke models of the logic EL of the form $\mathscr{M}_{EL} = \langle S, R_1, \ldots, R_n, \pi, \rangle$. Consider a time frame $(CK, <)$, where $CK = \langle t_0, t_1, t_2, \ldots \rangle$, and a function $v : CK \rightarrow \mathscr{K}^+$, mapping moments in time on the clock CK to a model in the class $\mathscr{K}^+$. A temporalised epistemic model is a tuple $\langle CK, <, v \rangle$, denoted by $\mathscr{M}_{TEL}$.

The semantics of TEL formulas are given as follows :

1. For $\phi \in \mathscr{L}_{EL}$, $\mathscr{M}_{TEL}, t_i \models \phi$ iff $\mathscr{M}_{EL} \models \phi$ where $v(t_i) = \mathscr{M}_{EL}$.
2. $\mathscr{M}_{TEL}, t_i \models \neg\phi$, iff $\mathscr{M}_{TEL}, t_i \not\models \phi$.
3. $\mathscr{M}_{TEL}, t_i \models \phi \wedge \psi$, iff $\mathscr{M}_{TEL}, t_i \models \phi$ and $\mathscr{M}_{TEL}, t_i \models \psi$.
4. $\mathscr{M}_{TEL}, t_i \models \mathbf{first}\ \phi$, iff $\mathscr{M}_{TEL}, t_0 \models \phi$.
5. $\mathscr{M}_{TEL}, t_i \models \mathbf{next}\ \phi$, iff $\mathscr{M}_{TEL}, t_{i+1} \models \phi$.

III. Labelled Tableaux Proof System for TEL

Tableaux are one of the favorite proof methods for modal logics. Various semantic tableau systems have been proposed in the literature [10], [11]. We adopt the labelled tableau system proposed by Governatori [10]. Formulas of the labelled tableaux system of TEL are in the form $\phi : l$, where ϕ is a formula of the logic and l is a label. The interpretation of $\phi : l$ is that ϕ is true at the world l.

A. Tableaux inference rules

The labelled tableaux system for the logic TEL consists of the following rules:

$$\frac{\phi \wedge \psi : s}{\substack{\phi : s \\ \psi : s}} \quad \frac{\neg(\phi \vee \psi) : s}{\substack{\neg\phi : s \\ \neg\psi : s}} \quad \frac{\neg(\phi \rightarrow \psi) : s}{\substack{\phi : s \\ \neg\psi : s}} \quad \frac{\phi \leftrightarrow \psi : s}{\substack{\phi \rightarrow \psi : s \\ \psi \rightarrow \phi : s}} \quad (C)$$

$$\frac{\substack{\phi \rightarrow \psi : s \\ \phi : w}}{\psi : s} \quad \frac{\substack{\phi \rightarrow \psi : s \\ \neg\psi : w}}{\neg\phi : s} \quad (I)$$

$$\frac{\phi \vee \psi : s}{\phi : s \mid \psi : s} \quad \frac{\neg(\phi \wedge \psi) : s}{\neg\phi : s \mid \neg\psi : s}$$

$$\frac{\phi \rightarrow \psi : s}{\neg\phi : s \mid \psi : s} \quad \frac{\neg(\phi \leftrightarrow \psi) : s}{\neg(\phi \rightarrow \psi) : s \mid \neg(\psi \rightarrow \phi) : s} \quad (D)$$

$$\frac{\forall x \phi(x) : s}{\phi(c) : s} \quad \frac{\neg\exists x \phi(x) : s}{\neg\phi(c) : s} \quad \frac{\exists x \phi(x) : s}{\phi(c) : s} \quad \frac{\neg\forall x \phi(x) : s}{\neg\phi(c) : s} \quad (Q)$$

Here C-rules are conjunctive rules; I-rules are implication rules; D-rules are disjunctive rules, which generate new branches; Q-rules are quantifier rules, in both universal and existential rules, x is a variable and c is a constant or any variable.

In the following, we present rules for modal operators.

$$\frac{\mathbf{first}\ \phi : s}{\phi : (t_0, s)} \quad \frac{\neg\ \mathbf{first}\ \phi : s}{\neg\phi : (t_0, s)}$$

$$\frac{\mathbf{next}\ \phi : s}{\phi : (t_i, s)} \quad \frac{\neg\mathbf{next}\ \phi : s}{\neg\phi : (t_i, s)}$$

$$\frac{\mathbf{next}\ \phi : (t_i, s)}{\phi : (t_{i+1}, s)} \quad \frac{\neg\mathbf{next}\ \phi : (t_i, s)}{\neg\phi : (t_{i+1}, s)} \quad (T)$$

For temporal rules, we need to introduce the concept of time-related world. A label may or may not relate to time. If a label does not relate to time, i.e., it does not contain a time reference, we say that it is a time-free reference, such as w; otherwise, it is a time-related reference, such as (t_i, s).

$$\frac{\mathbf{K}_a \phi : s_i}{\phi : (s_{i \rightarrow j}^a)} \quad \frac{\neg\mathbf{K}_a \phi : s_i}{\neg\phi : (s_{i \rightarrow j}^a)} \quad K$$

For knowledge rules, $s_{i \rightarrow j}^a$ represents the world accessible via R_a from s_i, where R_a is the accessibility relation associated with agent a.

$$\frac{\mathbf{K}_a \phi : (t_i, s_i)}{\phi : (t_i, s_{i \rightarrow j}^a)} \quad \frac{\neg\mathbf{K}_a \phi : (t_i, s_i)}{\neg\phi : (t_i, s_{i \rightarrow j}^a)} \quad (TE)$$

For temporal epistemic rules, the labels contain both temporal reference and world reference.

$$\frac{}{\phi : l \mid \neg\phi : l} \quad (Bi)$$

The bivalence rule represents the semantic counterpart of the cut rule of the sequent calculus, that is, a formula ϕ is either true or false in any given world. The bivalence rule is a zero-premise inference rule.

The following unification rule states that two labelled TEL formulas are complementary when the two formulas are complementary and their labels unify.

$$\frac{\substack{\phi(x) : l_n \\ \neg\phi(y) : l_m}}{\times} \quad (U)$$

Here, variables x and y unify as usual. For unifying world labels for TEL, we may adopt the label unification for a fibred logic of belief and time proposed by Orgun *et al* [12].

B. Logical Consequence and Proof Search

Logical consequence is the relation that holds between a set of formulas and a formula, we write:

$$G \models_{TEL} U \Rightarrow \phi,$$

which denotes that formula ϕ is derived from G and U, G is the set of global assumptions, and U is the set of local assumptions. ϕ is a logical consequence that follows from G and U.

We add the following assumption rules into the tableau proof system.

$$\frac{\phi \in G}{\phi : S} \quad \frac{\phi \in U}{\phi : s_i} \quad \frac{\phi : S}{\phi : s_i} \tag{A}$$

where S is variable which represents any state, and s_i is a specific state.

As usual with tableaux methods, proving of a formula ϕ of TEL is to construct a counter model for ϕ by assuming that formula ϕ is false in some TEL model. The TEL proof search procedure is as follows:

1. First we choose an open uncompleted branch $\mathscr{B}$, and apply the one-premise rules till the branch $\mathscr{B}$ is completed. If the resulting branch is not completed, go to 2.
2. We apply the two-premise rule till the resulting branch is completed. If the resulting branch is not completed, go to 3.
3. We apply the zero-premise rule till the resulting branch is completed. If the resulting branch is not completed, go to 4.
4. If there are two complementary formulas in a branch, we apply the unification rule, then the branch is closed.
5. The steps (1 - 4) are repeated for each branch generated by the bivalence rule.

In the following, we use a simple example to show how to reason about agent knowledge using the labelled tableau system.

Example - bit transmission protocol

A bit-transmission protocol [13] involves two agents, an initiator I, and a receiver R. First, I sends a bit to R, and continues to do so till it receives an acknowledgement from R. R sends an acknowledgement to I if it receives a bit. Therefore, if we have the following global assumptions:

(a) **next** $receive(I, Ack) \rightarrow receive(R, Bit)$.
(b) $receive(R, Bit) \wedge (Bit = n) \rightarrow \mathbf{K}_R (Bit = n)$.

and the following local assumptions:

(c) $receive(I, Ack)$.
(d) $Bit = n$.

Then we can prove the following formula:

(e) $\mathbf{K}_I \mathbf{K}_R (Bit = n)$.

The proof is given as follows.

1	$\neg \mathbf{K}_I \mathbf{K}_R (Bit = n) : s_0$	[$\neg$e]
2	$receive(I, Ack) : s_0$	[c]
3	$Bit = n : s_0$	[d]

4	**next** $receive(I, Ack) \rightarrow receive(R, Bit) : S$	[a]
5	$receive(R, Bit) \wedge (Bit = n) \rightarrow \mathbf{K}_R (Bit = n) : S$	[b]
6	$receive(R, Bit) \wedge (Bit = n) \rightarrow \mathbf{K}_R (Bit = n) : s_0$	[5Q&A]
7.1	$\neg(receive(R, Bit) \wedge (Bit = n)) : s_0$	[6D]
7.2	$\mathbf{K}_R (Bit = n) : s_0$	[6D]
8	$\neg \mathbf{K}_R (Bit = n)) : s_{0 \rightarrow j}^i$	[1K]
9	$\times$	{8/7.2}
10	$\neg receive(R, Bit) : s_0$	[7.1D]
11	$\neg (Bit = n) : s_0$	[7.1D]
12	$\times$	{11/3}
13	**next** $receive(I, Ack) \rightarrow receive(R, Bit) : s_0$	[4Q&A]
14.1	$\neg$**next** $receive(I, Ack) : s_0$	[13D]
14.2	$receive(R, Bit) : s_0$	[13D]
15	$\times$	{14.2/10}
16	$\neg\, receive(I, Ack) : (t_i, s_0)$	[14.1T]
17	$\times$	{16/2}
		$\square$

For proving the property, we begin with its negation, which is labelled by s_0 (a specific state). In the proof process, when we introduce a local assumption, we also label it by s_0; when we introduce a global assumption we label it by S. When a branch reaches a node with $\times$, then the branch is closed. We apply the unification rule without applying expansion rules in suitable cases. A closure of a node generates a closure for all branches that the node could have. This technique simplifies the proof procedure. This proof has all branches closed, so we have proved the formula (e).

C. Soundness and Completeness

The labelled tableau system presented in this paper is sound and complete. As usual with tableaux a proof of ϕ is a closed tableau for $\neg \phi$. A tableau system is sound and complete for a particular logic if it is able to generate a closed tableau for the negation of a valid formula, and open tableaux for all satisfiable formulas. For more details on the soundness and completeness of tableaux we refer the reader to the literature [10], [14].

IV. MODEL CHECKING FOR TEL

Model checking is used for checking whether a given structure is a model of a given logical formula. We now provide a model checking algorithm for TEL based on the technique proposed by Franceschet et al. [7].

Let **TL** be a temporal logic, **EL** be an epistemic logic, and **TEL** be the temporalised epistemic logic. To check whether a formula ϕ of a temporalised logic **TEL** is satisfied in a **TEL** model, we consider the set of the corresponding monolithic subformulas of ϕ in the logic **EL**. If we can determine the values of all the monolithic formulas in the corresponding **EL** model, then we will determine the truth value of ϕ in the **TEL** model.

A TEL model $\mathscr{M}_{TEL}$ is a tuple $\mathscr{M}_{TEL} = \langle ck, <, v \rangle$. Let $\mathscr{M}$ be a finite TEL model, and ϕ a TEL formula. We need to check whether there exists a state $t \in ck$ such that $\mathscr{M}, t \models_{TEL} \phi$. Let $\widehat{\Phi}$ be the set of maximal knowledge monolithic subformulas of ϕ, and $\widehat{\phi}$ be the TL formula obtained by substituting every formula $\alpha \in \widehat{\Phi}$ by a new proposition letter P.

For an example, given a TEL formula $\phi = \textbf{first next}(\gamma \wedge \mathbf{K}_{a_i}\ \psi)$, the formula $(\gamma \wedge \mathbf{K}_{a_i}\ \psi)$ has four subformulas:

1) $\gamma \wedge \mathbf{K}_{a_i}\ \psi$,
2) γ,
3) $\mathbf{K}_{a_i}\ \psi$,
4) ψ.

Here formula 2 is a monolithic formula; formula 4 is a monolithic formula, but it is not a maximal monolithic formula, because it is a subformula of formula 3. Therefore the set of maximal monolithic subformulas of ϕ is:

$$\widehat{\Phi} = \{\gamma, \mathbf{K}_{a_i}\ \psi\}$$

Replacing γ and $\mathbf{K}_{a_i}\psi$ by new propositional letters P and Q respectively, we obtain a temporal formula:

$$\widehat{\phi} = \textbf{first next}(P \wedge Q)$$

For model matching, time matching is an important issue. The global clock is the sequence of all natural numbers, that is, $\langle 0, 1, 2, \ldots \rangle$. A local clock is a subsequence of the global clock, it could be empty $\langle \rangle$, or finite $\langle t_0, t_1, t_2, \ldots, t_n \rangle$, or infinite $\langle t_0, t_1, t_2, \ldots \rangle$. Therefore for a given local clock $ck_i = \langle t_0, t_1, t_2, \ldots \rangle$, The rank of t_j on ck_i is j, we define that, $rank(ck_i, t_j) = j$. Let $n = rank(t_j, ck_i)$, if $\textbf{next first}\ \phi$ is true at the moment t_j. Then the following properties should be true.

- $\textbf{first}\ \phi$ is true in ck_i.
- ϕ is true at ck_i^0, ck_i^0 denotes the first time moment t_0 of the clock ck_i.
- $\textbf{first}\ \phi$ is true at ck_i^j, $0 \leq j \leq n$.
- $ck_i \models \textbf{first}\ \phi$, $0 \leq i \leq k$.

Let MC_{EL}, MC_{TL}, and MC_{TEL} be model checkers for the knowledge, temporal, and temporalised logics, respectively. Given a model checking instance, these programs will return a corresponding truth value.

We give the model checking procedure for TEL as follows:

> **Function** MC_{TEL}
> **Input**: a TEL model $\mathcal{M}_{TEL}$, and a TEL formula ϕ.
> **compute** $\widehat{\Phi}$ and $\widehat{\phi}$
> **for** every $t \in ck$
> $\pi(t) = \emptyset$
> **endfor**
> **for** every $\alpha \in \widehat{\Phi}$
>   **for** every $t \in ck$
>   **if** $\text{MC}_{EL}(\mathcal{M}_{EL}, \alpha) = true$
>   **then** $\pi(t) = \pi(t) \cup \{p\}$
>   **endfor**
> **endfor**
> **return** $\text{MC}_{TL}(\mathcal{M}_{TL}, \widehat{\phi})$.

Given a TEL model and TEL formula, MC_{TEL} computes the set $\widehat{\Phi}$ and the formula $\widehat{\phi}$ first. Then for every moment $t \in ck$, it initializes a valuation function π to the empty set. Then for every maximal monolithic formula $\alpha \in \widehat{\Phi}$ and every moment $t \in ck$, it invokes the model checker MC_{EL} on the input consisting of the EL model and the EL formula α. If the execution of MC_{EL} returns true, then the proposition p is added to $\pi(t)$. Finally, it invokes the model checker MC_{TL} on the input consisting of the temporal model and the formula $\widehat{\phi}$, and it returns the output.

With a finite TEL model and a TEL formula ϕ as input, we give the following definitions for function MC_{TEL}:

- **Termination:** MC_{TEL} returns either true or false.
- **Soundness:** If MC_{TEL} returns true, then for a existing $t \in ck$, $\mathcal{M}_{TEL}, t \models \phi$.
- **Completeness:** If MC_{TEL} returns false, then for every $t \in ck$, $\mathcal{M}_{TEL}, t \not\models \phi$.

V. Conclusion

We have proposed a labelled tableau system and a model checking method for a temporal epistemic logic called TEL. TEL is suitable for reasoning about the evolution of agent knowledge through time. Our approach could be useful in the designing, implementing and verifying agent systems. The model checking method is designed for automated theory verification; it is able to deal with both temporal and knowledge properties of systems. While there are many other proposed temporal epistemic logics, very few of them are the result of systematic combinations of temporal and epistemic logics. Future research includes investigating combinations of other modal and temporal logics for specifying agent systems, other combination techniques, and developing automated reasoning techniques and model checking methods for such logics.

References

[1] D. M. Gabbay, "Fibring logics." *Journal of Logic, Language and Information*, vol. 9, no. 4, pp. 511–513, 2000.
[2] M. Kracht and F. Wolter, "Properties of independently axiomatizable bimodal logics." in *The Journal of Symbolic Logic*, vol. 56(4), 1991, pp. 1469–1485.
[3] D. M. Gabbay and V. B. Shehtman, "Products of modal logics, part 1," *Logic Journal of the IGPL*, vol. 6, no. 1, pp. 73–146, 1998.
[4] M. Finger and D. M. Gabbay, "Adding a temporal dimension to a logic system." *Journal of Logic, Language and Information*, pp. 1:203–233, 1992.
[5] C. Liu, M. A. Ozols, and M. A. Orgun, "A temporalised belief logic for specifying the dynamics of trust for multi-agent systems." in *Proceedings of the Ninth Asian Computer Science Conference 2004*. LNCS, Vol. 3321, pp. 142–156. Springer, 2004.
[6] R. H. Bordini, M. Fisher, W. Visser, and M. Wooldridge, "Verifying multi-agent programs by model checking," *Autonomous Agents and Multi-Agent Systems*, vol. 12, no. 2, pp. 239–256, 2006.
[7] M. Franceschet, A. Montanari, and M. d. Rijke, "Model checking for combined logics with an application to mobile systems," *Autom. Softw. Eng.*, vol. 11, no. 3, pp. 289–321, 2004.
[8] A. F. Layouni, L. Logrippo, and K. J. Turner, "Conflict detection in call control using first-order logic model checking," in *ICFI*, 2007, pp. 66–82.
[9] C. Liu and M. A. Orgun, "Verification of reactive systems using temporal logic with clocks." *Theoretical Computer Science*, vol. 220, no. 2, pp. 377–408, 1999.
[10] G. Governatori, "Labelled tableaux for multi-modal logics," in *Proceedings of the 4th International Workshop on Theorem Proving with Analytic Tableaux and Related Methods*, 1995, pp. 79–94.
[11] K. Adi, M. Debbabi, and M. Mejri, "A new logic for electronic commerce protocols," *Theor. Comput. Sci.*, vol. 291, no. 3, pp. 223–283, 2003.
[12] M. A. Orgun, G. Governatori, and C. Liu, "Modal tableaux for verifying stream authentication protocols," *Autonomous Agents and Multi-Agent Systems*, vol. 19, no. 1, pp. 53–75, 2009.
[13] A. Lomuscio and M. J. Sergot, "The bit transmission problem revisited," in *AAMAS*, 2002, pp. 946–947.
[14] M. Fitting and R. L. Mendelsohn, *First-Order Modal Logic*. Kluwer Academic Publishers, 1999.