Tableau-based decision procedure for full coalitional multiagent temporal-epistemic logic of linear time

Valentin Goranko
School of Mathematics
University of the Witwatersrand
Johannesburg, South Africa
goranko@maths.wits.ac.za

Dmitry Shkatov
School of Computer Sciece
University of the Witwatersrand
Johannesburg, South Africa
dmitry@cs.wits.ac.za

ABSTRACT

We develop a tableau-based decision procedure for the full coalitional multiagent temporal-epistemic logic of linear time **CMATEL(CD+LT)**. It extends **LTL** with operators of common and distributed knowledge for all coalitions of agents. The tableau procedure runs in exponential time, matching the lower bound obtained by Halpern and Vardi for a fragment of our logic, thus providing a complexity-optimal decision procedure for **CMATEL(CD+LT)**.

General Terms

Theory

Keywords

Logics for multi-agent systems, decision procedures, tableaux

1. INTRODUCTION

Knowledge and time are among the most important aspects of multiagent systems. Various *temporal-epistemic logics*, proposed as logical frameworks for reasoning about these ascpects of multiagent systems were studied in a number of publications during the 1980's, eventually summarized in a uniform and comprehensive study by Halpern and Vardi [4]. In [4], the authors considered several essential characteristics of temporal-epistemic logics: *one vs. several agents, synchrony vs. asynchrony, (no) learning, (no) forgetting, linear vs. branching time*, and the (non-) existence of a *unique initial state*. Based on these, they identify and analyze 96 temporal-epistemic logics and obtain lower bounds for the complexity of a satisfiability problem in each of them. It turns out that most of the logics with more than one agent who do not learn or do not forget, are undecidable (with common knowledge), or decidable but with non-elementary time lower bound (without common knowledge). For the remaining multiagent logics, the lower bounds from [4] range from PSPACE (systems without common knowledge), through EXPTIME (with common knowledge), to EXPSPACE (synchronous systems with no learning and unique initial state). To the best of our knowledge, however, even for the logics from [4] with a relatively low

complexity lower bound, no decision procedures with matching upper bounds have been developed. In this paper, we set out to develop such decision procedures based on incremental tableaux, starting with the multiagent case over linear time, which involves no essential interaction between knowledge and time. It turns out that, under no other assumptions regarding learning or forgetting, the synchronous and asynchronous semantics are equivalent with respect to satisfiability. We consider a more expressive epistemic language than the ones considered in [4], to wit, the one involving operators for common and for distributed knowledge for *all coalitions* of agents. We call the resulting logic **CMATEL(CD+LT)** ("*Coalitional Multi-Agent Temporal Epistemic Logic with operators for Common and Distributed knowledge and Linear Time*"). The decision procedure for satisfiability in **CMATEL(CD+LT)** developed herein runs in exponential time, which together with the lower bound for the fragment of **CMATEL(CD+LT)** obtained in [4], implies EXPTIME-completeness of **CMATEL(CD+LT)**.

2. THE LOGIC CMATEL(CD+LT)

2.1 Syntax

The language $\mathcal{L}$ of **CMATEL(CD+LT)** contains a set `AP` of atomic propositions, a sufficient repertoire of Boolean connectives, say $\neg$ ("not") and $\wedge$ ("and"), the temporal operators $\bigcirc$ ("next") and $\mathcal{U}$ ("until") of the logic **LTL**, as well as the epistemic operators $\mathbf{D}_A \varphi$ ("*it is distributed knowledge among agents in A that φ*"), and $\mathbf{C}_A \varphi$ ("*it is common knowledge among agents of A that φ*") for every non-empty $A \subseteq \Sigma$, where Σ is the set of names of agents belonging to $\mathcal{L}$. The set Σ is assumed to be finite and non-empty; its subsets are called *coalitions* (of agents). Thus, the formulae of **CMATEL(CD+LT)** are defined as follows:

$$\varphi := p \mid \neg\varphi \mid (\varphi_1 \wedge \varphi_2) \mid \bigcirc\varphi \mid (\varphi_1 \, \mathcal{U} \varphi_2) \mid \mathbf{D}_A \varphi \mid \mathbf{C}_A \varphi$$

where p ranges over `AP` and A ranges over the set of non-empty subsets of Σ, henceforth denoted $\mathcal{P}^+(\Sigma)$. We write $\varphi \in \mathcal{L}$ to mean that φ is a formula of $\mathcal{L}$.

The operators of individual knowledge $\mathbf{K}_a \varphi$, where $a \in \Sigma$ ("*agent a knows that φ*"), can then be defined as $\mathbf{D}_{\{a\}}\varphi$, henceforth written $\mathbf{D}_a\varphi$. The other Boolean and temporal connectives can be defined as usual. We omit parentheses when this does not result in ambiguity.

Formulae of the form $\neg\mathbf{C}_A\varphi$ are *epistemic eventualities*, while those of the form $\varphi\,\mathcal{U}\psi$ are *temporal eventualities*.

$\mathbb{N} = \{0, 1, \ldots\}$ denotes the set of natural numbers.

2.2 Semantics

DEFINITION 2.1. *A temporal-epistemic system (TES) is a tuple* $\mathfrak{G} = (\Sigma, S, R, \{\mathcal{R}_A^D\}_{A \in \mathcal{P}^+(\Sigma)}, \{\mathcal{R}_A^C\}_{A \in \mathcal{P}^+(\Sigma)})$, *where:*

1. Σ *is a finite, non-empty set of* agents*;*
2. $S \neq \emptyset$ *is a set of* states*;*
3. *R is a non-empty set of* runs*; where each $r \in R$ is a function $r : \mathbb{N} \mapsto S$. A pair (r, n), where $r \in R$ and $n \in \mathbb{N}$, is called a* point*. The set of all points in $\mathfrak{G}$ is denoted $P(\mathfrak{G})$. Every point (r, n) represents the state $r(n)$; note, however, that different points can represent the same state.*
4. *for every $A \in \mathcal{P}^+(\Sigma)$, $\mathcal{R}_A^D$ and $\mathcal{R}_A^C$ are binary relation on $P(\mathfrak{G})$, such that $\mathcal{R}_A^C$ is the reflexive and transitive closure of $\bigcup_{A' \subseteq A} R_{A'}^D$.*

DEFINITION 2.2. *A temporal-epistemic frame (TEF) is a TES $\mathfrak{G} = (\Sigma, S, R, \{\mathcal{R}_A^D\}_{A \in \mathcal{P}^+(\Sigma)}, \{\mathcal{R}_A^C\}_{A \in \mathcal{P}^+(\Sigma)})$, where each $\mathcal{R}_A^D$ is an equivalence relation satisfying the following condition: (†) $\mathcal{R}_A^D = \bigcap_{a \in A} \mathcal{R}_{\{a\}}^D$. If condition (†) is replaced by the following: (††) $\mathcal{R}_A^D \subseteq \mathcal{R}_B^D$ whenever $B \subseteq A$, then $\mathfrak{F}$ is a temporal-epistemic pseudo-frame (pseudo-TEF).*

Notice that, in (pseudo-)TEFs, $\mathcal{R}_A^C$ is the transitive closure of $\bigcup_{a \in A} R_{\{a\}}^D$ and, thus, an equivalence relation.

DEFINITION 2.3. *A temporal-epistemic model (TEM, for short) is a tuple $\mathcal{M} = (\mathfrak{F}, L)$, where*

(i) $\mathfrak{F}$ *is a TEF with a set of runs R;*
(ii) $L : R \times \mathbb{N} \mapsto \mathcal{P}(\mathtt{AP})$ *is a* labeling function*, where $L(r, n)$ is the set of atomic propositions true at (r, n).*

If the condition (i) is replaced by the requirement that $\mathfrak{F}$ is a pseudo-TEF, then $\mathcal{M}$ is a temporal-epistemic pseudo-model (pseudo-TEM)*.*

A TES $\mathfrak{G}$ is called *synchronous* if for every $A \in \mathcal{P}^+(\Sigma)$, if $((r, n), (r', n')) \in \mathcal{R}_A^D$, then $n = n'$. Synchronous temporal-epistemic (pseudo)-models are defined accordingly. Hereafter we consider the general case, but all definitions and results apply likewise to the synchronous case, unless stated otherwise. The tableau construction can accommodate the synchronous case at no extra cost and eventually we show that, under no other assumptions, the presence or absence of synchrony does not affect the satisfiability of formulae.

DEFINITION 2.4. *The satisfaction of formulae at points in (pseudo-)TEMs is defined as follows:*
$\mathcal{M}, (r, n) \Vdash p$ *iff* $p \in L(r, n)$;
$\mathcal{M}, (r, n) \Vdash \neg\varphi$ *iff not* $\mathcal{M}, (r, n) \Vdash \varphi$;
$\mathcal{M}, (r, n) \Vdash \varphi \wedge \psi$ *iff* $\mathcal{M}, (r, n) \Vdash \varphi$ *and* $\mathcal{M}, (r, n) \Vdash \psi$;
$\mathcal{M}, (r, n) \Vdash \bigcirc\varphi$ *iff* $\mathcal{M}, (r, n+1) \Vdash \varphi$;
$\mathcal{M}, (r, n) \Vdash \varphi\mathcal{U}\psi$ *iff* $\mathcal{M}, (r, i) \Vdash \psi$ *for some* $i \geq n$
        such that $\mathcal{M}, (r, j) \Vdash \varphi$ *for every* $n \leq j < i$;
$\mathcal{M}, (r, n) \Vdash \mathbf{D}_A\varphi$ *iff* $\mathcal{M}, (r', n') \Vdash \varphi$
        for every $((r, n), (r', n')) \in \mathcal{R}_A^D$;
$\mathcal{M}, (r, n) \Vdash \mathbf{C}_A\varphi$ *iff* $\mathcal{M}, (r', n') \Vdash \varphi$
        for every $((r, n), (r', n')) \in \mathcal{R}_A^C$;

Note, that in the semantics defined above *the labelling function acts on points, not states*, i.e., it is *point-based*. To make the semantics *state-based*, one must impose the additional condition: if $r(n) = r'(n')$ then $L(r, n) = L(r', n')$.

However, for the case of linear time logics these two semantics are equivalent in terms of satisfiability and validity (this is an easy consequence of the fact that, in the linear case, all epistemic operators have built-in implicit universal quantification over paths).

The satisfaction condition for the operator $\mathbf{C}_A$ can be paraphrased in terms of reachability. Let $\mathfrak{F}$ be a (pseudo-) TEF over the set of runs R and let $(r, n) \in R \times \mathbb{N}$. We say that a point (r', n') is *A-reachable from* (r, n) if either $r = r'$ and $n = n'$ or there exists a sequence $(r, n) = (r_0, n_0), (r_1, n_1), \ldots, (r_{m-1}, n_{m-1}), (r_m, n_m) = (r', n')$ of points in $R \times \mathbb{N}$ such that, for every $0 \leq i < m$, there exists $a_i \in A$ such that $((r_i, n_i), (r_{i+1}, n_{i+1})) \in R_{a_i}^D$. Then, the satisfaction condition for $\mathbf{C}_A$ becomes equivalent to the following:
$\mathcal{M}, (r, n) \Vdash \mathbf{C}_A\varphi$ iff $\mathcal{M}, (r', n') \Vdash \varphi$ whenever (r', n') is A-reachable from (r, n).

Satisfiability and validity in (a class of) models is defined as usual.

It is easy to see that if $\Sigma = \{a\}$, then $\mathbf{D}_a\varphi \leftrightarrow \mathbf{C}_a\varphi$ is valid in every TEM for every $\varphi \in \mathcal{L}$. Thus, the single-agent case is essentially trivialized and, therefore, we assume hereafter that Σ contains at least 2 (names of) agents.

3. HINTIKKA STRUCTURES

Even though we are ultimately interested in testing formulae of $\mathcal{L}$ for satisfiability in a TEM, the tableau procedure we present tests for satisfiability in a more general kind of semantic structures, namely a *Hintikka structure*. We will show that $\theta \in \mathcal{L}$ is satisfiable in a TEM iff it is satisfiable in a Hintikka structure, hence the latter test is equivalent to the former. The advantage of working with Hintikka structures lies in the fact that they contain as much semantic information about θ as is necessary, and no more. More precisely, while models provide the truth value of every formula of $\mathcal{L}$ at every state, Hintikka structures only determine the truth of formulae directly involved in the evaluation of a fixed formula θ, in whose satisfiability we are interested. Another important difference between models and Hintikka structures is that, in Hintikka structures the epistemic relations $\mathcal{R}_A^D$ and $\mathcal{R}_A^C$ only have to satisfy the properties laid down in Definition 2.1. All the other information about the desirable properties of epistemic relations is contained in the labeling of states in Hintikka structures. This labeling ensures that every Hintikka structure generates a pseudo-model (by the construction of Lemma 3.5), which can then be turned into a model using the construction of Lemma 3.9.

DEFINITION 3.1. *A set $\Delta \subseteq \mathcal{L}$ is* fully expanded *if it satisfies the following conditions ($\mathsf{Sub}(\psi)$ stands for the set of subformulae of ψ):*

1. *if* $\neg\neg\varphi \in \Delta$ *then* $\varphi \in \Delta$;
2. *if* $\varphi \wedge \psi \in \Delta$, *then* $\varphi \in \Delta$ *and* $\psi \in \Delta$;
3. *if* $\neg(\varphi \wedge \psi) \in \Delta$ *then* $\neg\varphi \in \Delta$ *or* $\neg\varphi \in \Delta$;
4. *if* $\neg\bigcirc\varphi \in \Delta$ *then* $\bigcirc\neg\varphi \in \Delta$;
5. *if* $\varphi\mathcal{U}\psi \in \Delta$ *then* $\psi \in \Delta$ *or* $\varphi, \bigcirc(\varphi\mathcal{U}\psi) \in \Delta$;
6. *if* $\neg(\varphi\mathcal{U}\psi) \in \Delta$ *then* $\neg\psi, \neg\varphi \in \Delta$ *or* $\neg\psi, \neg\bigcirc(\varphi\mathcal{U}\psi) \in \Delta$;
7. *if* $\mathbf{D}_A\varphi \in \Delta$ *then* $\mathbf{D}_{A'}\varphi \in \Delta$ *for every A' such that* $A \subseteq A' \subseteq \Sigma$;
8. *if* $\mathbf{D}_A\varphi \in \Delta$ *then* $\varphi \in \Delta$;
9. *if* $\mathbf{C}_A\varphi \in \Delta$ *then* $\mathbf{D}_a(\varphi \wedge \mathbf{C}_A\varphi) \in \Delta$ *for every $a \in A$;*

10. *if* $\neg\mathbf{C}_A\varphi \in \Delta$ *then* $\neg\mathbf{D}_a(\varphi \wedge \mathbf{C}_A\varphi) \in \Delta$ *for some* $a \in A$;
11. *if* $\psi \in \Delta$ *and* $\mathbf{D}_A\varphi \in \mathsf{Sub}(\psi)$ *then either* $\mathbf{D}_A\varphi \in \Delta$ *or* $\neg\mathbf{D}_A\varphi \in \Delta$.

DEFINITION 3.2. *A temporal-epistemic Hintikka structure (TEHS) is a tuple* $(\Sigma, S, R, \{\mathcal{R}_A^D\}_{A\in\mathcal{P}^+(\Sigma)}, \{\mathcal{R}_A^C\}_{A\in\mathcal{P}(\Sigma)}, H)$ *such that* $(\Sigma, S, R, \{\mathcal{R}_A^D\}_{A\in\mathcal{P}^+(\Sigma)}, \{\mathcal{R}_A^C\}_{A\in\mathcal{P}(\Sigma)})$ *is a TES, and H is a labeling of points in $R \times \mathbb{N}$ with sets of formulae, satisfying the following conditions, for all $(r,n) \in R \times \mathbb{N}$:*

H1 *if* $\neg\bigcirc\varphi \in H(r,n)$, *then* $\varphi \notin H(r,n)$;

H2 $H(r,n)$ *is fully expanded*;

H3 *if* $\bigcirc\varphi \in H(r,n)$, *then* $\varphi \in H(r,n+1)$;

H4 *if* $\varphi\mathcal{U}\psi \in H(r,n)$, *then there exists* $i \geq n$ *such that* $\psi \in H(r,i)$ *and* $\varphi \in H(r,j)$ *holds for every* $n \leq j < i$;

H5 *if* $\neg\mathbf{D}_A\varphi \in H(r,n)$, *then there exists* $r' \in R$ *and* $n' \in \mathbb{N}$ *such that* $((r,n),(r',n')) \in \mathcal{R}_A^D$ *and* $\neg\varphi \in H(r',n')$;

H6 *if* $((r,n),(r',n')) \in \mathcal{R}_A^D$, *then* $\mathbf{D}_{A'}\varphi \in H(r,n)$ *iff* $\mathbf{D}_{A'}\varphi \in H(r',n')$, *for every* $A' \subseteq A$;

H7 *if* $\neg\mathbf{C}_A\varphi \in H(r,n)$, *then there exists* $r' \in R$ *and* $n' \in \mathbb{N}$ *such that* $((r,n),(r',n')) \in \mathcal{R}_A^C$ *and* $\neg\varphi \in H(r',n')$.

Synchronous TEHSs (STEHSs) are defined likewise.

DEFINITION 3.3. *A set of formulae* Θ *is* satisfiable *in a TEHS $\mathcal{H}$ with a labeling function H if there exists a point $(r,n) \in \mathcal{H}$ such that $\Theta \subseteq H(r,n)$. Analogously for formulae.*

Now, we show that $\theta \in \mathcal{L}$ is satisfiable in a TEM iff it is satisfiable in a TEHS. One direction is almost immediate, as every TEM naturally induces a TEHS. More precisely, given a TEM $\mathcal{M}$, define the *extended labeling* L^+ on the set of points of $\mathcal{M}$ as follows: $L^+(r,n) = \{\varphi \mid \mathcal{M},(r,n) \Vdash \varphi\}$ for every (r,n). The following claim is then straightforward.

LEMMA 3.4. *Let* $\mathcal{M} = (\mathfrak{F}, L)$ *be a TEM satisfying* $\theta \in \mathcal{L}$, *and let L^+ be the extended labeling on $\mathcal{M}$. Then, $\mathcal{H} = (\mathfrak{F}, \mathsf{AP}, L^+)$ is a TEHS satisfying θ.*

For the opposite direction, we first prove that the existence of a TEHS satisfying θ implies the existence of a pseudo-model satisfying θ; then, we show that this in turn implies the existence of a model satisfying θ.

LEMMA 3.5. *Let* $\theta \in \mathcal{L}$ *be such that there exists a TEHS for θ. Then, θ is satisfiable in a pseudo-TEM.*

PROOF. Let $\mathcal{H} = (\Sigma, S, R, \{\mathcal{R}_A^D\}_{A\in\mathcal{P}^+(\Sigma)}, \{\mathcal{R}_A^C\}_{A\in\mathcal{P}(\Sigma)}, H)$ be a TEHS for θ. We build a pseudo-TEM satisfying θ as follows. First, for every $A \in \mathcal{P}^+(\Sigma)$, let $\mathcal{R}_A'^D$ be the reflexive, symmetric, and transitive closure of $\bigcup_{A\subseteq B}\mathcal{R}_B^D$ and let $\mathcal{R}_A'^C$ be the transitive closure of $\bigcup_{a\in A}\mathcal{R}_a'^D$. Notice that $\mathcal{R}_A^D \subseteq \mathcal{R}_A'^D$ and $\mathcal{R}_A^C \subseteq \mathcal{R}_A'^C$ for every $A \in \mathcal{P}^+(\Sigma)$. Next, let $L(r,n) = H(r,n) \cap \mathsf{AP}$, for every point $(r,n) \in R \times \mathbb{N}$. It is then easy to check that $\mathcal{M}' = (\Sigma, S, R, \{\mathcal{R}_A'^D\}_{A\in\mathcal{P}^+(\Sigma)}, \{\mathcal{R}_A'^C\}_{A\in\mathcal{P}^+(\Sigma)}, \mathsf{AP}, L)$ is a pseudo-TEM. It is also easy to check that the construction preserves synchrony.

To complete the proof of the lemma, we show, by induction on the formula $\chi \in \mathcal{L}$ that, for every point (r,n) and every $\chi \in \mathcal{L}$, the following hold:

(i) $\chi \in H(r,n)$ implies $\mathcal{M}',(r,n) \Vdash \chi$;

(ii) $\neg\chi \in H(r,n)$ implies $\mathcal{M}',(r,n) \Vdash \neg\chi$.

Let χ be some $p \in \mathsf{AP}$. Then, $p \in H(r,n)$ implies $p \in L(r,n)$ and thus, $\mathcal{M}',(r,n) \Vdash p$; if, on the other hand, $\neg p \in$

$H(r,n)$, then due to (H1), $p \notin H(r,n)$ and thus $p \notin L(r,n)$; hence, $\mathcal{M}',(r,n) \Vdash \neg p$.

Assume that the claim holds for all subformulae of χ; then, we have to prove that it holds for χ, as well.

Suppose that $\chi = \neg\varphi$. If $\neg\varphi \in H(r,n)$, then the inductive hypothesis immediately gives us $\mathcal{M}',(r,n) \Vdash \neg\varphi$; if, on the other hand, $\neg\neg\varphi \in H(r,n)$, then by virtue of (H2), $\varphi \in H(r,n)$ and hence, by inductive hypothesis, $\mathcal{M}',(r,n) \Vdash \varphi$ and thus $\mathcal{M}',(r,n) \Vdash \neg\neg\varphi$.

The cases of $\chi = \varphi \wedge \psi$ and $\chi = \bigcirc\varphi$ are straightforward, using (H2) and (H3).

Suppose that $\chi = \mathbf{D}_A\varphi$. Assume, first, that $\mathbf{D}_A\varphi \in H(r,n)$. In view of the inductive hypothesis, it suffices to show that $((r,n),(r',n')) \in \mathcal{R}_A'^D$ implies $\varphi \in H(r,n)$. Assuming $((r,n),(r',n')) \in \mathcal{R}_A'^D$, there are two cases to consider. If $(r,n) = (r',n')$, then the conclusion immediately follows from (H2). Otherwise, there exists an undirected path from (r,n) to (r',n') along the relations $\mathcal{R}_{A'}^D$, where each A' is a superset of A. Then, due to (H6), $\mathbf{D}_A\varphi \in H(r',n')$; hence, by (H2), $\varphi \in H(r',n')$, as desired.

Now, let $\neg\mathbf{D}_A\varphi \in H(r,n)$. By (H5), there exist $r' \in R$ and $n' \in \mathbb{N}$ such that $((r,n),(r',n')) \in \mathcal{R}_A^D$ and $\neg\varphi \in H(r',n')$. As $\mathcal{R}_A^D \subseteq \mathcal{R}_A'^D$, the claim follows from the inductive hypothesis.

Suppose that $\chi = \mathbf{C}_A\varphi$. Assume that $\mathbf{C}_A\varphi \in H(r,n)$. By inductive hypothesis, it suffices to show that if (r',n') is A-reachable from (r,n), then $\varphi \in H(r',n')$. If $(r,n) = (r',n')$ the claim follows from (H2). So, suppose for some $m \geq 1$, there exists a sequence of points $(r,n) = (r_0,n_0),\ldots,(r_{m-1},n_{m-1}),(r_m,n_m) = (r',n')$ such that, for every $0 \leq i < m$, there exists $a_i \in A$ such that $((r_i,n_i),(r_{i+1},n_{i+1})) \in \mathcal{R}_{a_i}'^D$. Then, for every $0 \leq i < m$, there exists $a_i \in A$ such that $((r_i,n_i),(r_{i+1},n_{i+1})) \in R_{a_i}$. We can then show by induction on i, using (H2) and (H6), that $\mathbf{C}_A\varphi \in H(r_i,n_i)$ holds for every $0 \leq< m$; hence, $\mathbf{D}_{a_i}(\varphi \wedge \mathbf{C}_A\varphi) \in H(r_i,n_i)$. Therefore, $\varphi \in H(r_{i+1},n_{i+1})$ by (H2) and (H6). By taking $i = m-1$ we obtain $\varphi \in H(r',n')$, as required.

Now, assume $\neg\mathbf{C}_A\varphi \in H(r,n)$. Then, the claim follows from (H7) and the inductive hypothesis, since $\mathcal{R}_A^C \subseteq \mathcal{R}_A'^C$.

Suppose that $\chi = \varphi\mathcal{U}\psi$. If $\varphi\mathcal{U}\psi \in H(r,n)$, then the conclusion immediately follows from (H4) and the inductive hypothesis. Suppose, on the other hand, that $\neg(\varphi\mathcal{U}\psi) \in H(r,n)$. Then, by (H2), $\neg\psi,\neg\varphi \in H(r,n)$ or $\neg\psi,\bigcirc\neg(\varphi\mathcal{U}\psi) \in H(r,n)$. In case former case, the inductive hypothesis immediately gives us the desired result. In the latter, inductive hypothesis gives us $\mathcal{M},(r,n) \nVdash \psi$ and (H3) gives us $\neg(\varphi\mathcal{U}\psi) \in H(r,n+1)$. Now the argument can be repeated. Ultimately, using inductive hypothesis, we either get a finite path $(r,n),\ldots,(r,i)$ such that $\mathcal{M},(r,i) \nVdash \varphi$ and $\mathcal{M},(r,j) \nVdash \psi$ holds for all $n \leq j \leq i$, or we get an infinite path $(r,n),(r,n+1),\ldots$ such that $\mathcal{M},(r,i) \nVdash \psi$ for all $i \geq n$. In either case, $\mathcal{M},(r,n) \nVdash \psi\mathcal{U}\psi$. $\square$

To show that satisfiability of a formula in a pseudo-TEM implies its satisfiability in a TEM, we use a modification of the construction from [1, Appendix A1] (see also [6]).

DEFINITION 3.6. *Let* $\mathcal{M} = (\Sigma, S, R, \{\mathcal{R}_A^D\}_{A\in\mathcal{P}^+(\Sigma)}, \{\mathcal{R}_A^C\}_{A\in\mathcal{P}^+(\Sigma)}, \mathsf{AP}, L)$ *be a (pseudo-)TEM and let* $r,r' \in R$ *and* $n,n' \in \mathbb{N}$. *A maximal path from (r,n) to (r',n') in $\mathcal{M}$ is a sequence* $(r,n) = (r_0,n_0),A_0,(r_1,n_1),\ldots,A_{m-1},(r_m,n_m) = (r',n')$ *such that, for every* $0 \leq i < m$, $((r_i,n_i),(r_{i+1},n_{i+1})) \in \mathcal{R}_{A_i}^D$, *but* $((r_i,n_i),(r_{i+1},n_{i+1})) \notin \mathcal{R}_B^D$ *for any B such that*

$A_i \subset B \subseteq \Sigma$. A segment ρ' of a maximal path ρ starting and ending with a point is a sub-path of ρ.

DEFINITION 3.7. *Let* $\rho = (r_0, n_0), A_0 \dots, A_{n-1}, (r_m, n_m)$ *be a maximal path in* $\mathcal{M}$. *The* reduction *of* ρ *is obtained by, first, replacing in* ρ *every longest sub-path* $(r_p, n_p), A_p, (r_{p+1}, n_{p+1}) \dots, A_{p+q-1}, (r_{p+q}, n_{p+q})$ *such that* $r_p = r_{p+1} = \dots = r_{p+q}$ *with* r_p *(i.e., eliminating loops) and, then, by replacing in the resultant path every longest sub-path* $(r_j, n_j), A_j, (r_{j+1}, n_{j+1}) \dots, A_{j+m-1}, (r_{j+m}, n_{j+m})$ *such that* $A_j = A_{j+1} = \dots = A_{j+m-1}$ *with* $(r_j, n_j), A_j, (r_{j+m}, n_{j+m})$ *(reducing multiple transitions along the same relation into a single transition). A maximal path is* reduced *if it equals its reduction.*

DEFINITION 3.8. *A (pseudo-)TEM* $\mathcal{M}$ *is* forest-like *if, for every* $r, r' \in R$ *and every* $n, n' \in \mathbb{N}$, *there exists at most one reduced maximal path from* (r, n) *to* (r', n').

One difference of the construction presented below from the one in [1, Appendix A1] is that, instead of producing a tree-like model, we rather produce a forest-like one, partly since every "temporal level" of the model we are going to build will not be connected by epistemic relations to any other temporal level, and partly because even within a single temporal level we will, in general, construct more than one "epistemic tree".

LEMMA 3.9. *If* $\theta \in \mathcal{L}$ *is satisfiable in a (synchronous) pseudo-TEM, then it is satisfiable in a (synchronous) forest-like TEM.*

PROOF. We will only consider the synchronous case, as it requires extra care. Suppose that θ is satisfied in a synchronous pseudo-TEM $\mathcal{M} = (\Sigma, S, R, \{\mathcal{R}_A^D\}_{A \in \mathcal{P}^+(\Sigma)}, \{\mathcal{R}_A^C\}_{A \in \mathcal{P}^+(\Sigma)}, \mathtt{AP}, L)$ at a point (r, n). To build a synchronous forest-like TEM $\mathcal{M}'$ satisfying θ, we use the modified tree-unraveling technique. First, every "epistemic tree" within a temporal level of the model will be made up of all *maximal* paths, rather than all paths, as in the standard tree-unraveling, since we want to ensure that paths between points are unique with respect to the relations $\mathcal{R}_A^D$ indexed by maximal coalitions, which will allow us to fix "defects" with respect to the D-relations. Second, every level will, in general, be made up of more than one epistemic tree, as every point at level $m \neq 0$ created as part of temporal run starting at a level $k < m$, will be a root of a separate tree.

The construction starts by taking a submodel $\mathcal{M}_{(r,n)}$ of $\mathcal{M}$ generated by the point $x = (r, n)$ at which θ is satisfiable.

Next, we define $\mathcal{M}'$ by recursion on the temporal levels. We view a level k as partitioned into clusters $\{S_1^k, S_2^k, \dots\}$, such that if $((r, k), (r', k)) \in S_i^k$, there is an (undirected) path along D-relations between (r, k) and (r', k).

We start from level 0, corresponding to level n in $\mathcal{M}$ and level 0 in $\mathcal{M}_{(r,n)}$. This level contains only one cluster S^0, generated by point x. In general, however, a level k will have more than one cluster, so we describe the construction in more general terms. At level k, for each cluster S_i^k, we choose arbitrarily a point $(r_i, k) \in S_i^k$ (at level 0, however, we choose x); this point is going to be the root of an epistemic tree associated with that cluster. Now, we call a maximal path ρ in $\mathcal{M}$ a (r_i, k)-*max-path* if the first component of ρ is (r_i, k). We denote the last element of ρ by $l(\rho)$. Notice that (r_i, k) is by itself an (r_i, k)-max-path. Now, let $\widehat{S}_i^k$ be the set of all (r_i, k)-max-paths in $\mathcal{M}$. For every $A \in \mathcal{P}^+(\Sigma)$,

let $\mathcal{R}_A^{*D} = \{ (\rho, \rho') \mid \rho, \rho' \in \bigcup_i \widehat{S}_i^k$ and $\rho' = \rho, A, l(\rho') \}$. Let, furthermore, $\mathcal{R}_A'^D$ to be the reflexive, symmetric, and transitive closure of $\mathcal{R}_A^{*D}$. Notice that $(\rho, \rho') \in \mathcal{R}_A'^D$ holds iff one of the paths ρ and ρ' extends the other by a sequence of A-steps. Therefore, two different states in $\bigcup_i \widehat{S}_i^k$ can only connected by $\mathcal{R}_A'^D$ for at most one maximal coalition A. Further, we stipulate the following *downwards closure condition*: whenever $(\rho, \tau) \in \mathcal{R}_A'^D$ and $B \subseteq A$, then $(\rho, \tau) \in \mathcal{R}_B'^D$. The relations $\mathcal{R}_A'^C$ are then defined as in any TEF.

We next describe how to create level $m+1$ of $\mathcal{M}'$ assuming that level m has already been defined. First, carry out for $m + 1$ the construction described in the previous paragraph for an arbitrary level k. Secondly, for every pair of states $\rho \in \bigcup_i \widehat{S}_i^m$ and $\tau \in \bigcup_i \widehat{S}_i^{m+1}$ make $(\tau, m + 1)$ a temporal successor of (ρ, m) if $l(\tau)$ is a such successor of $l(\rho)$ in $\mathcal{M}$.

To complete the definition of $\mathcal{M}'$, we put $L'(\rho) = L(l(\rho))$ for every $\rho \in P(\mathcal{M}')$, where $P(\mathcal{M}')$ is the set of points of $\mathcal{M}'$. It is clear from the construction, namely from the downward saturation condition above, that $\mathcal{M}'$ is a synchronous pseudo-TEM. We now show that it is a TEM satisfying θ.

To prove the first part of the claim, we need extra terminology. We call a maximal path $\rho_1, A_1, \rho_2, \dots, A_{n-1}, \rho_n$ in $\mathcal{M}'$ *primitive* if, for every $0 \leq i < n$, either $(\rho_i, \rho_{i+1}) \in \mathcal{R}_{A_i}^{*D}$ or $(\rho_{i+1}, \rho_i) \in \mathcal{R}_{A_i}^{*D}$. A primitive path $\rho_1, A_1, \rho_2, \dots, A_{n-1}, \rho_n$ is *non-redundant* if there is no $0 \leq i < n$ such that $\rho_i = \rho_{i+2}$ and $A_i = A_{i+1}$. Intuitively, in a non-redundant path we never go from a state ρ (forward or backward) along a relation and then immediately back to ρ along the same relation. Since the relations $\mathcal{R}_A^{*D}$ are edges of a tree, it immediately follows that (S' denotes the state space of $\mathcal{M}'$):

(‡) for every pair of states $\rho, \tau \in S'$, there exists at most one non-redundant primitive path from ρ to τ.

Lastly, we call a primitive path $\rho_1, A, \rho_2, \dots, A, \rho_n$ an A-*primitive path*.

We will now show that maximal reduced paths in $\mathcal{M}'$ stand in one-to-one correspondence with non-redundant primitive paths. It will then follow from (‡) that maximal reduced paths between any two states of $\mathcal{M}'$ are unique, and thus $\mathcal{M}'$ is forest-like, as claimed. Let $P = \rho_1, A_1, \dots, A_{n-1}, \rho_n$, where $\rho_1 = \rho$ and $\rho_n = \tau$, be a maximal reduced path from ρ to τ in $\mathcal{M}'$. Since $(\rho_i, \rho_{i+1}) \in \mathcal{R}_{A_i}'^D$, there exists a non-redundant A_i-primitive path from ρ_i to ρ_{i+1}, which in view of (‡) is unique. Let us obtain a path P' from ρ to τ by replacing in ρ every link $(\rho_i, A_i, \rho_{i+1})$ by the corresponding non-redundant A_i-primitive path from ρ_i to ρ_{i+1}. Call P' an *expansion* of P. In view of (‡), every path has a unique expansion. Now, it is easy to see that P is a reduction of P'. Since the reduction of a given path is unique, too, it follows that there exists a one-to-one correspondence between reduced paths and non-redundant primitive paths in $\mathcal{M}'$.

We now prove that $\mathcal{R}_A'^D = \bigcap_{a \in A} \mathcal{R}_a'^D$ for every $A \in \mathcal{P}^+(\Sigma)$, and hence $\mathcal{M}'$ is a TEM. The left to right inclusion is immediate, as $\mathcal{M}'$ is pseudo-TEM. For the other direction, assume that $((r, n), (r', n)) \in \mathcal{R}_a'^D$ holds for every $a \in A$. Then, for every $a \in A$, there exists a path, and therefore a maximal reduced path, from (r, n) to (r', n) along relations $\mathcal{R}_{A'}'^D$ such that $a \in A'$. As $\mathcal{M}'$ is forest-like, there is only one maximal reduced path from (r, n) to (r', n). Therefore, the relations $\mathcal{R}_{A'}^D$ linking (r, n) to (r', n) along this path are such that $A \subseteq A'$ for every A'. Then, by the downwards closure condition, there is a path from (r, n) to (r', n) along the relation $\mathcal{R}_A'^D$ and, hence, $((r, n), (r', n)) \in \mathcal{R}_A'^D$, as desired.

Finally, it remains to prove that $\mathcal{M}'$ satisfies θ. First, notice that $(\rho, \tau) \in \mathcal{R}'_A$ iff there exists an A-primitive path from ρ to τ. Hence, as every $\mathcal{R}_A$ is an equivalence relation, if $(\rho, \tau) \in \mathcal{R}'_A$, then $(l(\rho), l(\tau)) \in \mathcal{R}'_A$. It is now straightforward to check that the relation $Z = \{ (\rho, l(\rho) \mid \rho \in S' \}$ is a bisimulation between $\mathcal{M}'$ and $\mathcal{M}$. Since $(x, l(x)) \in Z$, it follows that $\mathcal{M}', x \Vdash \theta$, and we are done. $\square$

THEOREM 3.10. *Let $\theta \in \mathcal{L}$. Then, θ is satisfiable in a TEM iff there exists a TEHS satisfying θ.*

PROOF. Immediate from Lemmas 3.4, 3.5 and 3.9. $\square$

4. TABLEAUX FOR CMATEL(CD + LT)

In the present section, we describe the tableau procedure for testing formulae of **CMATEL(CD+LT)** for satisfiability in synchronous systems, as this case requires more care. We then briefly mention how to modify the procedure for asynchronous case and argue the the output of both procedures for the same input formula is the same, implying the equivalence of two semantics.

4.1 Overview of the tableau procedure

The tableau procedure for testing a formula $\theta \in \mathcal{L}$ for satisfiability attempts to construct a non-empty graph $\mathcal{T}^\theta$ (called *tableau*), whose nodes are finite subsets of $\mathcal{L}$, representing *sufficiently many* TEHSs, in the sense that, if θ is satisfiable in a TEHS, it is satisfiable in a one represented by a tableau for θ. The philosophy underlying our tableau algorithm is essentially the same as the one underpinning the tableau procedure for **LTL** from [7], recently adapted to multiagent epistemic logics in [2]; this philosophy can be traced back to [5]. To make the present paper self-contained, we outline the basic ideas behind our tableau algorithm in line with those references. The particulars of the tableaux presented here, however, are specific to **CMATEL(CD+LT)**.

Usually, tableaux work by decomposing the input formula into simpler formulae. In the classical propositional case, "simpler" implies shorter, thus ensuring the termination of the procedure. The decomposition into simpler formulae in the tableau for classical propositional logic produces a tree representing an exhaustive search for a Hintikka set (the classical analogue of Hintikka structures) for the input formula θ. If at least one leaf of that tree produces a Hintikka set for θ, the search has succeeded and θ is pronounced satisfiable; otherwise it is declared unsatisfiable.

When applied to logics containing fixpoint-definable operators, such as $\mathbf{C}_A$ and $\mathcal{U}$, these two defining features of the classical tableau method no longer apply. First, the decomposition of fixpoint formulae, which is done by unfolding their fixpoint definitions, produces larger formulae: $\mathbf{C}_A\varphi$ is decomposed into formulae of the form $\mathbf{D}_a(\varphi \wedge \mathbf{C}_A\varphi)$, while $\varphi \mathcal{U} \psi$ is decomposed into ψ and $\varphi \wedge \bigcirc(\varphi \mathcal{U} \psi)$. Hence, we need a termination-ensuring mechanism. In our tableaux, such a mechanism is provided by the use (and reuse) of so called "prestates", whose role is to ensure the finiteness of the construction and, hence, termination of the procedure. Second, the only reason why a tableau may fail to produce a Hintikka set for the input formula in the classical case is that every attempt to build such a set results in a collection of formulae containing a *patent inconsistency*, i.e., a complementary pair of formulae $\varphi, \neg\varphi$. In the case of **CMATEL(CD+LT)**, there are other such reasons, as the

tableaux in this case are meant to represent TEHSs, which are more involved structures than classical Hintikka sets. One additional reason has to do with eventualities: the presence of an eventuality $\neg\mathbf{C}_A\varphi$ in the label of a state s of a TEHS $\mathcal{H}$ requires the existence in $\mathcal{H}$ of an A-path from s to a state t whose label contains $\neg\varphi$ (condition (H7) of Definition 3.2). An analogous requirement applies to eventualities of the form $\varphi \mathcal{U} \psi$ due to condition (H4) of Definition 3.2. The tableau analogs of these conditions is called *realization of eventualities*. If a tableau contains nodes with unrealized eventualities, then it cannot produce a TEHS, and thus it is "bad". The third possible reason for a tableau to be "bad" has to do with successor nodes: it may so happen that some of the successors of a node s which are necessary for the satisfaction of s are unsatisfiable. Notice that TEHSs, and consequently the associated tableaux, contain two kinds of "successor" nodes: temporal and epistemic. The non-satisfiability of either kind of successor can ruin the chances of a tableau node to correspond to a state of a TEHS.

The tableau procedure consists of three major phases: *pretableau construction*, *prestate elimination*, and *state elimination*. During the pretableau construction phase, we produce a directed graph $\mathcal{P}^\theta$—called the *pretableau* for θ—whose set of nodes properly contains the set of nodes of the tableau $\mathcal{T}^\theta$ we are building. The nodes of $\mathcal{P}^\theta$ are sets of formulae of two kinds: *states* and *prestates*. States are fully expanded sets, meant to represent (labels of) states of a Hintikka structure, while prestates play a temporary role in the construction of $\mathcal{T}^\theta$. During the prestate elimination phase, we create a smaller graph $\mathcal{T}_0^\theta$ out of $\mathcal{P}^\theta$, called the *initial tableau for θ*, by eliminating all the prestates from $\mathcal{P}^\theta$ and accordingly redirecting its edges. Finally, during the state elimination phase, we remove from $\mathcal{T}_0^\theta$ all the states, if any, that cannot be satisfied in a TEHS, either because they contain unrealized eventualities or because they lack a necessary successor (patently inconsistent states are removed "on the fly" during the state creation stage). The elimination procedure results in a (possibly empty) subgraph $\mathcal{T}^\theta$ of $\mathcal{T}_0^\theta$, called the *final tableau for θ*. If some state Δ of $\mathcal{T}^\theta$ contains θ, we declare θ satisfiable; otherwise, we declare it unsatisfiable. The construction of the tableau is illustrated in Example 1 given at the end of Section 4.4.

4.2 Pretableau construction phase

All states and prestates of the pretableau $\mathcal{P}^\theta$ constructed during this phase are "time-stamped", the notation $\Gamma^{[n]}$ indicating that prestate Γ was created as the nth component of a run; analogously for states.

The pretableau contains three types of edge, described below. As already mentioned, a tableau attempts to produce a compact representation of a sufficient number of TEHSs for the input formula, which are the result of an exhaustive search for a TEHS satisfying θ. One type of edge, depicted by unmarked double arrows $\Longrightarrow$, represents the search dimension of the tableau. Exhaustive search considers all possible alternatives, which arise when expanding prestates into states by branching when dealing with the "disjunctive formulae". Thus, when we draw a double arrow from a prestate Γ to states Δ and Δ' (depicted as $\Gamma \Longrightarrow \Delta$ and $\Gamma \Longrightarrow \Delta'$, respectively), this intuitively means that, in any TEHS, a state whose label extends the set Γ has to contain at least one of Δ and Δ'. Our first construction rule, **(SR)**, prescribes how to create tableau states from prestates.

Given a set $\Gamma \subseteq \mathcal{L}$, we say that Δ is a *minimal fully expanded extension of* Γ if Δ is fully expanded, $\Gamma \subseteq \Delta$, and there is no Δ' such that $\Gamma \subseteq \Delta' \subset \Delta$ and Δ' is fully expanded.

Rule (SR) Given a prestate $\Gamma^{[n]}$ such that **(SR)** has not been applied to **(SR)** earlier, do the following:

1. Add all minimal fully expanded extensions $\Delta^{[n]}$ of $\Gamma^{[n]}$ that are not patently inconsistent as *states*;
2. if $\Delta^{[n]}$ contains no formulae $\bigcirc\varphi$, add $\bigcirc\top$ to it;
3. for each so obtained state $\Delta^{[n]}$, put $\Gamma^{[n]} \Longrightarrow \Delta^{[n]}$;
4. if, however, the pretableau already contains a state $\Delta'^{[m]}$ that coincides with $\Delta^{[n]}$, do not create another copy of $\Delta'^{[m]}$, but only put $\Gamma^{[n]} \Longrightarrow \Delta'^{[m]}$.

We denote by **states**$(\Gamma^{[n]})$ the set of states $\{ \Delta \mid \Gamma^{[n]} \Longrightarrow \Delta \}$. Note that we remove patently inconsistent states "on the fly", thus never making them part of a pretableau.

Notice that in all construction rules, as in **(SR)**, we allow reuse of (pre)states, which were originally stamped with a possibly different time-stamp. This does not correspond to one state or prestate being part of two different runs, at different moments of time (the absolute time is supposed to be the same in all runs, even though agents may not be able to observe it, in asynchronous systems); rather, the "futures" of these runs, starting from the reused (pre)state can be assumed to be identical, modulo the time difference.

The second type of edge in a pretableau represents epistemic relations in the TEHSs that the procedure attempts to build. This type of edge is represented by single arrows marked with epistemic formulae whose presence in the source state requires the presence in the tableau of a target state, reachable by a particular epistemic relation. All such formulae have the form $\neg\mathbf{D}_A\varphi$ (as can be seen from Definition 3.2). Intuitively if, say $\neg\mathbf{D}_A\varphi \in \Delta^{[n]}$, then we need some prestate $\Gamma^{[n]}$ containing $\neg\varphi$ to be accessible from $\Delta^{[n]}$ by $\mathcal{R}_A^D$ (notice that the newly created prestates bear the same time stamp as the source state; this reflects the fact that we are considering the synchronous case). The reason we mark these single arrows not just by a coalition A, but by a formula $\neg\mathbf{D}_A\varphi$, is that we have to remember not just what relation connects states whose labels contain $\Delta^{[n]}$ and $\Gamma^{[n]}$, but why we had to create this particular $\Gamma^{[n]}$. This information will be needed when we start eliminating prestates, and then states. We now formulate the rule producing this second type of edges in the pretableau.

Rule (DR): Given a state $\Delta^{[n]}$ such that $\neg\mathbf{D}_A\varphi \in \Delta^{[n]}$, $\Delta^{[n]}$ and **(DR)** has not been applied to $\Delta^{[n]}$ earlier, do the following:

1. Create a new prestate $\Gamma^{[n]} = \{\neg\varphi\} \cup \bigcup_{A' \subseteq A}\{ \mathbf{D}_{A'}\psi \mid \mathbf{D}_{A'}\psi \in \Delta^{[n]} \} \cup \bigcup_{A' \subseteq A}\{ \neg\mathbf{D}_{A'}\psi \mid \neg\mathbf{D}_{A'}\psi \in \Delta^{[n]} \}$;
2. connect $\Delta^{[n]}$ to $\Gamma^{[n]}$ with $\xrightarrow{\neg\mathbf{D}_A\varphi}$;
3. if, however, the tableau already contains a prestate $\Gamma'^{[n]} = \Gamma^{[n]}$, do not add another copy of $\Gamma'^{[n]}$, but simply connect $\Delta^{[n]}$ to $\Gamma'^{[n]}$ with $\xrightarrow{\neg\mathbf{D}_A\varphi}$.

Lastly, the third type of edge, depicted by single unmarked arrow $\longrightarrow$, represents temporal transitions. We now state the rule that creates such arrows.

Rule (Next): Given a state $\Delta^{[n]}$ such that **(Next)** has not been applied to $\Delta^{[n]}$ earlier, do the following:

1. Create a new prestate $\Gamma^{[n+1]} = \{ \varphi \mid \bigcirc\varphi \in \Delta^{[n]} \}$;

2. connect $\Delta^{[n]}$ to $\Gamma^{[n+1]}$ with $\longrightarrow$;
3. if, however, the tableau already contains a prestate $\Gamma'^{[m]} = \Gamma^{[n+1]}$, do not add another copy of $\Gamma'^{[m]}$, but simply connect $\Delta^{[n]}$ to $\Gamma'^{[m]}$ with $\longrightarrow$.

Note that, due to step 2 in **(SR)**, every state contains at least one formula of the form $\bigcirc\varphi$.

Having stated the rules, we now describe how the construction phase works. We start off by creating a single prestate $\{\theta\}$, where θ is the input formula. Then we alternatingly apply **(DR)** and **(Next)** to the prestates created at the previous stage and then applying **(SR)** to the newly created states. The construction state is over when the applications of **(DR)** and **(Next)** do not produce any new prestates.

4.3 Prestate elimination phase

At this phase we remove from $\mathcal{P}^\theta$ all the prestates and double arrows, by applying the following rule:

Rule (PR) For every prestate Γ in $\mathcal{P}^\theta$, do the following:

1. Remove Γ from $\mathcal{P}^\theta$;
2. if there is a state Δ in $\mathcal{P}^\theta$ with $\Delta \xrightarrow{\chi} \Gamma$, then for every state $\Delta' \in \mathbf{states}(\Gamma)$, put $\Delta \xrightarrow{\chi} \Delta'$;
3. if there is a state Δ in $\mathcal{P}^\theta$ with $\Delta \longrightarrow \Gamma$, then for every state $\Delta' \in \mathbf{states}(\Gamma)$, put $\Delta \longrightarrow \Delta'$.

The resulting graph, denoted $\mathcal{T}_0^\theta$, is called the *initial tableau*.

4.4 State elimination phase

During this phase we remove from $\mathcal{T}_0^\theta$ states that are not satisfiable in a TEHS. There are two reasons why a state Δ of $\mathcal{T}_0^\theta$ can turn out to be unsatisfiable: either satisfiability of Δ requires satisfiability of some other (epistemic or temporal) successor states which are unsatisfiable, or Δ contains an eventuality that is not realized in the tableau. Accordingly, we have three elimination rules (as two different rules deal with epistemic and temporal successors): **(E1E)**, **(E1T)**, and **(E2)**.

Formally, the state elimination phase is divided into stages; we start at stage 0 with $\mathcal{T}_0^\theta$; at stage $n + 1$ we remove from the tableau $\mathcal{T}_n^\theta$ obtained at the previous stage exactly one state, by applying one of the elimination rules, thus obtaining the tableau $\mathcal{T}_{n+1}^\theta$. We state the rules below, where S_m^θ denotes the set of states of $\mathcal{T}_m^\theta$.

(E1E) If $\Delta \in S_n^\theta$ contains a formula $\chi = \neg\mathbf{D}_A\varphi$ and $\Delta \xrightarrow{\chi} \Delta'$ does not hold for any $\Delta' \in S_n^\theta$, obtain $\mathcal{T}_{n+1}^\theta$ by eliminating Δ from $\mathcal{T}_n^\theta$.

(E1T) If If $\Delta \in S_n^\theta$ and $\Delta \longrightarrow \Delta'$ does not hold for any $\Delta' \in S_n^\theta$, obtain $\mathcal{T}_{n+1}^\theta$ by eliminating Δ from $\mathcal{T}_n^\theta$.

For the third elimination rule, we need the concept of *eventuality realization*. We say that the eventuality $\neg\mathbf{C}_A\varphi$ is realized at Δ in $\mathcal{T}_n^\theta$ if there exists a finite path $\Delta = \Delta_0, \Delta_1, \ldots, \Delta_m$ (where $m \geq 0$) such that $\neg\varphi \in \Delta_m$ and for every $0 \leq i < m$ there exist $\chi_i = \mathbf{D}_B\psi_i$ such that $B \subseteq A$ and $\Delta_i \xrightarrow{\chi_i} \Delta_{i+1}$. Analogously, we say that the eventuality $\varphi\,\mathcal{U}\psi$ is realized at Δ in $\mathcal{T}_n^\theta$ if there exists a finite path $\Delta = \Delta_0, \Delta_1, \ldots, \Delta_m$ (where $m \geq 0$) such that $\psi \in \Delta_m$, and for every $0 \leq i < m$, both $\Delta_i \longrightarrow \Delta_{i+1}$ and $\varphi \in \Delta_i$ hold.

(E2) If $\Delta \in S_n^\theta$ contains a (temporal or epistemic) eventuality ξ that is not realized at Δ in $\mathcal{T}_n^\theta$, then obtain $\mathcal{T}_{n+1}^\theta$ by removing Δ from $\mathcal{T}_n^\theta$.

We check for realization of eventualities by running the following iterative procedure that eventually marks all states that realize a given eventuality ξ in $\mathcal{T}_n^\theta$. If $\xi = \neg\mathbf{C}_A\varphi$, then initially, we mark all $\Delta \in S_n^\theta$ such that $\neg\varphi \in \Delta$. Then, we repeat the following procedure until no more states get marked: for every still unmarked $\Delta \in S_n^\theta$, mark Δ if there is at least one Δ' such that $\Delta \overset{\mathbf{D}_B\psi}{\longrightarrow} \Delta'$ for some $B \subseteq A$ and Δ' is marked. The procedure for eventualities of the form $\varphi\,\mathcal{U}\,\psi$ is analogous.

We have so far described individual rules and their implementation; to describe the state elimination phase as a whole, we need to specify the order of their application. We need to be careful, as having applied **(E2)**, we could have removed all the states accessible from some Δ either along the arrows marked with an epistemic formula χ or along unmarked arrows $\longrightarrow$; hence, we need to reapply **(E1E)** and **(E1T)** to the resultant tableau to remove such Δ's. Conversely, having applied **(E1E)** and **(E1T)**, we could have thrown away some states that were needed for realizing certain eventualities; hence, we need to reapply **(E2)**. Therefore, we need to apply **(E2)**, **(E1E)**, and **(E1T)** in a dovetailed sequence that cycles through all the eventualities. More precisely, we arrange all eventualities occurring in $\mathcal{T}_0^\theta$ in a list $\xi_1, \ldots, \xi_m$. Then, we proceed in cycles. Each cycle consists of alternatingly applying **(E2)** to the pending eventuality (starting with ξ_1), and then applying **(E1E)** and **(E1T)** to the resulting tableau, until all the eventualities have been dealt with, i.e., we reached ξ_m. These cycles are repeated until no state is removed in a whole cycle. Then, the state elimination phase is over.

The graph produced at the end of the state elimination phase is called the *final tableau for θ*, denoted by $\mathcal{T}^\theta$ and its set of states is denoted by S^θ.

DEFINITION 4.1. *The final tableau $\mathcal{T}^\theta$ is open if $\theta \in \Delta$ for some $\Delta \in S^\theta$; otherwise, $\mathcal{T}^\theta$ is closed.*

The tableau procedure returns "no" if the final tableau is closed; otherwise, it returns "yes" and, moreover, provides sufficient information for producing a finite pseudo-model satisfying θ; that construction is sketched in Section 5.

EXAMPLE 1. *In this example, we show how our procedure works on the formula $\neg\mathbf{C}_{\{a,b\}}p\,\mathcal{U}\,\mathbf{D}_{\{a,c\}}p$. Below is the complete pretableau for this formula.*
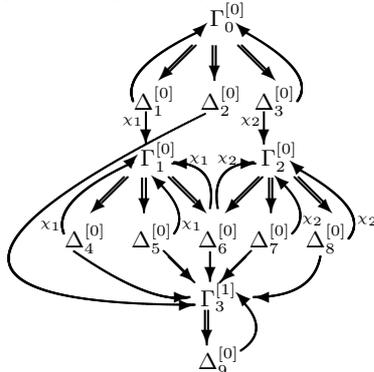
$\chi_1 = \neg\mathbf{D}_a(p \wedge \mathbf{C}_{\{a,b\}}p);\ \chi_2 = \neg\mathbf{D}_b(p \wedge \mathbf{C}_{\{a,b\}}p);$
$\Gamma_0 = \{\neg\mathbf{C}_{\{a,b\}}p\,\mathcal{U}\,\mathbf{D}_{\{a,c\}}p = \theta\};$
$\Delta_1 = \{\theta, \neg\mathbf{C}_{\{a,b\}}p, \bigcirc\theta, \chi_1\};\ \Delta_2 = \{\mathbf{D}_{\{a,c\}}p, p, \bigcirc\top\};$
$\Delta_3 = \{\theta, \neg\mathbf{C}_{\{a,b\}}p, \bigcirc\theta, \chi_2\};\ \Gamma_1 = \{\chi_1, \neg(p \wedge \mathbf{C}_{\{a,c\}}p)\};$

$\Gamma_2 = \{\chi_2, \neg(p \wedge \mathbf{C}_{\{a,c\}}p)\};\ \Delta_4 = \{\chi_1, \neg p, \bigcirc\top\};$
$\Delta_5 = \{\chi_1, \neg\mathbf{C}_{\{a,b\}}p, \bigcirc\top\};\ \Delta_6 = \{\chi_1, \neg\mathbf{C}_{\{a,b\}}p, \chi_2, \bigcirc\top\};$
$\Delta_7 = \{\chi_2, \neg\mathbf{C}_{\{a,c\}}p, \bigcirc\top\};\ \Delta_8 = \{\chi_2, \neg p, \bigcirc\top\};$
$\Gamma_3 = \{\top\};\ \Delta_9 = \{\top, \bigcirc\top\}.$

The initial tableau is obtained by removing all prestates (the Γs) and redirecting the arrows (i.e, Δ_1 will be connected by unmarked single arrows to itself, Δ_2, and Δ_3). It is easy to check that no states get removed during the state elimination stage; hence, the tableau is open and θ is satisfiable.

We now briefly mention how to modify the above procedure for the asynchronous case. The only difference occurs in the **(DR)** rule: we now longer require that prestates produced during the application of this rule to a given state $\Delta^{[n]}$ should have the same time stamp as Δ (namely, n). A brief analysis of the procedure shows that this modification does not change the outcome of the procedure for a given formula. This, in particular, implies that the satisfiability-wise equivalence of synchronous and asynchronous semantics.

5. SOUNDNESS, COMPLETENESS, AND COMPLEXITY

The *soundness* of a tableau procedure amounts to claiming that if the input formula θ is satisfiable, then the tableau for θ is open. To establish soundness of the overall procedure, we use a series of lemmas showing that every rule by itself is sound; the soundness of the overall procedure is then an easy consequence. The proofs of the following three lemmas are straightforward.

LEMMA 5.1. *Let Γ be a prestate of $\mathcal{P}^\theta$ such that $\mathcal{M}, (r, n) \Vdash \Gamma$ for some TEM $\mathcal{M}$ and point (r, n). Then, $\mathcal{M}, (r, n) \Vdash \Delta$ holds for at least one $\Delta \in \mathbf{states}(\Gamma)$.*

LEMMA 5.2. *Let $\Delta \in S_m^\theta$, for $m \geq 0$, be such that $\mathcal{M}, (r, n) \Vdash \Delta$ for some TEM $\mathcal{M}$ and point (r, n), and let $\neg\mathbf{D}_A\varphi \in \Delta$. Then, there exists a point $(r', n') \in \mathcal{M}$ such that $((r, n), (r', n')) \in \mathcal{R}_A^D$ and $\mathcal{M}, (r', n') \Vdash \Delta'$ where $\Delta' = \{\neg\varphi\} \cup \bigcup_{A' \subseteq A}\{\mathbf{D}_{A'}\psi \mid \mathbf{D}_{A'}\psi \in \Delta\} \cup \bigcup_{A' \subseteq A}\{\neg\mathbf{D}_{A'}\psi \mid \neg\mathbf{D}_{A'}\psi \in \Delta\}.$*

LEMMA 5.3. *Let $\Delta \in S_m^\theta$, for $m \geq 0$, be such that $\mathcal{M}, (r, n) \Vdash \Delta$ for some TEM $\mathcal{M}$ and a point (r, n). Then, $\mathcal{M}, (r, n + 1) \Vdash \bigcirc(\Delta)$ where $\bigcirc(\Delta) = \{\varphi \mid \bigcirc\varphi \in \Delta\}.$*

LEMMA 5.4. *Let $\Delta \in S_m^\theta$, for $m \geq 0$, be such that $\mathcal{M}, (r, n) \Vdash \Delta$ for some TEM $\mathcal{M}$ and a point (r, n), and let $\neg\mathbf{C}_A\varphi \in \Delta$. Then, $\neg\mathbf{C}_A\varphi$ is realized at Δ in $\mathcal{T}_m^\theta$.*

Proof idea. Since $\neg\mathbf{C}_A\varphi$ is true at s, there is a path in $\mathcal{M}$ from s leading to a state satisfying $\neg\varphi$. Since the tableau performs exhaustive search, a chain of tableau states corresponding to those states in the model will be produced. $\square$

The next lemma is proved likewise.

LEMMA 5.5. *Let $\Delta \in S_m^\theta$, for $m \geq 0$, be such that $\mathcal{M}, (r, n) \Vdash \Delta$ for some TEM $\mathcal{M}$ and a point (r, n), and let $\varphi\,\mathcal{U}\,\psi \in \Delta$. Then, $\varphi\,\mathcal{U}\,\psi$ is realized at Δ in $\mathcal{T}_m^\theta$.*

THEOREM 5.6. *If $\theta \in \mathcal{L}$ is satisfiable in a TEM, then $\mathcal{T}^\theta$ is open.*

Proof sketch. Using the preceding lemmas, we show by induction on the number of stages in the state elimination phase that no satisfiable state can be eliminated due to any of the elimination rules. The claim then follows from Lemma 5.1. □

The *completeness* of a tableau procedure means that if the tableau for a formula θ is open, then θ is satisfiable in a TEM. In view of Theorem 3.10, it suffices to show that an open tableau for θ can be turned into a TEHS for θ.

LEMMA 5.7. *If $\mathcal{T}^\theta$ is open, then a (synchronous) TEHS for θ exists.*

Proof sketch. The TEHS $\mathcal{H}$ for θ is built by induction on the temporal levels, in order to take care of synchrony. The main concern is to ensure that all eventualities in the resultant structure are realized (all other properties of Hintikka structures easily transfer from an open tableau). We alternate between realizing epistemic eventualities (formulae of the form $\neg \mathbf{C}_A \varphi$) and temporal eventualities (formulae of the form $\varphi \mathcal{U} \psi$).

We start by building the 0th level of our prospective Hintikka structure from the level 0 of the open tableau. For each state $\Delta^{[0]}$ on this level, if $\Delta^{[0]}$ does not contain any epistemic eventualities, we define epistemic component for $\Delta^{[0]}$ to be $\Delta^{[0]}$ with exactly one successor reachable by $\neg \mathbf{D}_A \psi$, for each $\neg \mathbf{D}_A \psi \in \Delta^{[0]}$; if, on the other hand, $\neg \mathbf{C}_A \varphi \in \Delta^{[0]}$, then such a component is a tree obtained from a path in the tableau realizing $\neg \mathbf{C}_A \varphi$ at $\Delta^{[0]}$ by giving each component of the path "enough" successors, as described above. We recursively repeat the procedure extending the current tree by attaching to its leaves associated components. As all the unrealized epistemic eventualities are propagated down the components (hence, appear in the leaves of the tree), we can stitch them up together to obtain a structure in which epistemic eventuality is realized.

Now, having built the 0th level of our prospective Hintikka structure, we take care of realizing all the temporal eventualities contained in the states of level 0. This is done exactly as in the completeness proof of the tableau procedure for **LTL**: we define the temporal component for each $\Delta^{[0]}$ as follows: if $\Delta^{[0]}$ does not contain any temporal eventualities, then we take $\Delta^{[0]}$ with one of its temporal successors; otherwise, we take a temporal path realizing $\varphi \mathcal{U} \psi \in \Delta^{[0]}$. As eventualities are again passed down, we can stitch up an infinite, or ultimately periodic, path realizing all the eventualities contained in the states making up the path.

Next, we repeat the procedure inductively. For the mth epistemic level, we independently apply to each state on this level the procedure described above for level 0, so that "epistemic structures" unfolding from any two points on level m are disjoint, and also give to each newly created point a "history" consisting of a path of $m-1$ states of the form $\{\top\}$ (so that we do not create any new epistemic eventualities at the levels we have already "processed"). Having fixed all the epistemic eventualities at the mth level, we repeat the procedure described in the previous paragraph to fix all the temporal eventualities contained in states of level m.

Thus, we produce a chain of structures ordered by inclusion. Eventually, we take the (infinite) union of all the structures defined at the finite states of that construction, and then put $H(\Delta^{[n]}) = \Delta^{[n]}$ for every $\Delta^{[n]}$, to obtain a TEHS for θ. □

THEOREM 5.8 (COMPLETENESS). *Let $\theta \in \mathcal{L}$ and let $\mathcal{T}^\theta$ be open. Then, θ is satisfiable.*

PROOF. Immediate from Lemma 5.7 and Theorem 3.10. □

As for complexity, for lack of space, we only state that it runs within exponential time (the calculation is routine). Therefore, the **CMATEL(CD+LT)**-satisfiability is in EXPTIME, which together with the EXPTIME-hardness result from [4], implies that it is EXPTIME-complete.

6. CONCLUDING REMARKS

We developed an incremental-tableau based decision procedure for the full coalitional multiagent temporal-epistemic logic of linear time **CMATEL(CD+LT)**. In this case, there is no essential interaction between the temporal and the epistemic dimensions, which makes the tableau construction easier to build and less expensive to run, by reducing it to a combination of tableaux for **LTL** and for the (epistemic) logic **CMAEL(CD)** developed in [2]. We are convinced that our procedure is—besides being rather intuitive—practically much more efficient than the top-down tableaux, e.g., developed for a fragment of our logic in [3], and hence better suited to both manual and automated execution. It is also easily amenable to modifications suited to reasoning about subclasses of distributed systems, e.g., those with a unique initial state. The branching time case, which will be considered in a sequel to this paper, is essentially a combination of tableaux for **CTL** with those for **CMAEL(CD)**. On the other hand, the development of tableau-based procedures for those logics from [4] whose satisfiability problem has EXPSPACE lower bound is an open challenge.

7. REFERENCES

[1] R. Fagin, J. Y. Halpern, and M. Y. Vardi. What can machines know? On the properties of knowledge in distributed systems. *Journal of the ACM*, 39(2):328–376, April 1992.

[2] V. Goranko and D. Shkatov. Tableau-based procedure for deciding satisfiability in the full coalitional multiagent epistemic logic. In S. Artemov and A. Nerode, editors, *Proc. of the Symposium on Logical Foundations of Computer Science (LFCS 2009)*, volume 5407 of *Lecture Notes in Computer Science*, pages 197–213. Springer-Verlag, 2009.

[3] J. Y. Halpern and Y. Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial Intelligence*, 54:319–379, 1992.

[4] J. Y. Halpern and M. Y. Vardi. The complexity of reasoning about knowledge and time I: Lower bounds. *Journal of Computer and System Sciences*, 38(1):195–237, 1989.

[5] V. R. Pratt. A near optimal method for reasoning about action. *Journal of Computer and System Sciences*, 20:231–254, 1980.

[6] W. van der Hoek and J.-J. C. Meyer. Making some issues of implicit knowledge explicit. *International Journal of Foundations of Computer Science*, 3(2):193–224, 1992.

[7] P. Wolper. The tableau method for temporal logic: an overview. *Logique et Analyse*, 28(110–111):119–136, 1985.