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

Valentin Goranko Dmitry Shkatov

Abstract

We develop a sound and complete tableau-based decision procedure for the full coalitional multiagent temporal-epistemic logic of branching time ***CMATEL(CD+BT)*** *that extends logic* ***CTL*** *with epistemic operators for common and distributed knowledge for all coalitions of agents referred to in the language. The procedure runs in exponential time, which matches the lower bound established by Halpern and Vardi for a fragment of our logic, thus providing a complexity-optimal decision procedure and a complete deductive system for our logic.*

1 Introduction

Reasoning about knowledge and time is crucial for designing, and verifying properties of, distributed and multiagent systems. A number of *temporal-epistemic logics* were proposed as logical frameworks for modeling of, and reasoning about, these aspects of distributed systems in the 1980's. This research was summarized in the comprehensive study by Halpern and Vardi [7]. In that study, the authors considered several essential characteristics of temporal-epistemic logics, namely: *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 the satisfiability problem in all of them. It turns out that most of the systems with more than one agents who do not learn or do not forget are undecidable, often Π^1_1-hard (with common knowledge), or decidable but with non-elementary time lower bound (without common knowledge). For most of the remaining logics, the lower bounds established in [7] for the multiagent case range from PSPACE (synchronous systems without common knowledge), through EXPTIME (with common knowledge), to EXPSPACE (synchronous systems with no learning and unique initial state).

Despite the conceptual importance and wide range of potential applications of temporal-epistemic logics, to the best of our knowledge, no efficient decision procedures for logics studied in [7] had been developed until quite recently, even for the systems with a relatively low known lower bounds. The only exception is [11], where a top-down tableau-style decision procedure for the logic **ATEL**, which subsumes the basic branching-time logic considered in [7] and this paper, was presented. In our view (to be explained further), however, [11] should be seen as a contribution to the complexity-theoretic analysis of the temporal-epistemic logics rather than to the development of efficient decision procedures for them.

In the recent precursor [5] to the present paper, we set out to fill in the above-mentioned gap by developing a practically efficient (within the theoretical complexity bounds) tableau-based decision procedure for the coalitional multiagent temporal-epistemic logic of linear time **CMATEL(CD+LT)** (for both the synchronous and asynchronous cases), building both on Wolper's incremental tableaux for **LTL** [13] and on our earlier work on tableaux for the full coalitional multiagent (purely) epistemic logic **CMAEL(CD)** [6].

In the present paper, we report on the second, and final, part of the project undertaken with the publication of [5]; namely, we present a sound, complete, and terminating incremental tableau for the ***C**oalitional **M**ulti-**A**gent **T**emporal **E**pistemic **L**ogic with operators for **C**ommon and **D**istributed knowledge and **B**ranching **T**ime*, **CMATEL(CD+BT)**. The tableau procedure presented herein follows the tableau-building philosophy developed for the logics **PDL** by Pratt in [9], **UB** by Ben-Ari, Manna and Pnueli in [1], and **CTL** by Emerson and Halpern in [3]. Our procedure essentially combines incremental tableaux for **CTL** from the latter (see also [10] for a recent detailed exposition) and tableaux for the full coalitional multiagent epistemic logic **CMAEL(CD)** developed in [6]. In the present paper, as in [6], we work with a more expressive epistemic language than the one considered in [7], as it contains operators for common and distributed knowledge for *all nonempty coalitions* (i.e., subsets) of the set of agents. The resulting decision procedure for testing satisfiability in **CMA-**

TEL(CD+BT) runs in exponential time, which is the optimal lower-bound, as established in [11].

We should mention that, even though the procedure presented in [11] can be used to test **CMATEL(CD+BT)**-formulae for satisfiability, this would not give us the optimal procedure, since such a procedure would *always* require exponential time predicted by the worst-case estimate. The incremental tableau presented in this paper, on the other hand, on average requires much less time than the theoretical upper bound (this claim cannot be made mathematically precise without an a priori probability distribution on formulae; however, it is substantiated by example in [4], where we compare the incremental tableaux presented in that paper with the top-down tableux-style procedure from [12]).

Besides presenting the tableau-based procedure for **CMATEL(CD+BT)**, the other major objective of this paper is to demonstrate how two tableau procedures for logics with non-interacting fixed-point operators (the epistemic and the temporal ones, in our case) can be combined into a tableau procedure for the fusion of these logics, thus offering a contribution to the area of combination of logics (see e.g., [8]).

The present paper is structured as follows. In Section 2, we introduce the logic **CMATEL(CD+BT)**. In Section 3, we introduce Hintikka structures for **CMATEL(CD+BT)** and show that satisfiability of **CMATEL(CD+BT)**-formulae in Hintikka structures is equivalent to satisfiability in models introduced in Section 2. In Section 4, we present the tableau procedure for **CMATEL(CD+BT)** and, in Section 5, sketch out the proofs of soundness and completeness and briefly discuss the complexity of the procedure. The Appendix contains an example of a run of the procedure.

2 Syntax and semantics of the logic CMATEL(CD+BT)

The language $\mathcal{L}$ of **CMATEL(CD+BT)** contains a (possibly infinite) set **AP** of atomic propositions; the Boolean connectives $\neg$ ("not") and $\wedge$ ("and"); the unary temporal operators $\exists\bigcirc$ and $\forall\bigcirc$ (existential and universal "next", respectively); the binary temporal operators $\exists(-\,\mathcal{U}-)$ and $\forall(-\,\mathcal{U}-)$ (existential and universal "until", respectively), as well as the unary 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 finite, non-empty set of names of agents belonging to $\mathcal{L}$. Subsets of Σ are called *coalitions*. Thus, the formulae of $\mathcal{L}$ are defined as follows:

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

where p ranges over **AP** and A ranges over the set $\mathcal{P}^+(\Sigma)$ of non-empty subsets of Σ. We write $\varphi \in \mathcal{L}$ to mean that φ is a formula of $\mathcal{L}$ and $\Delta \subseteq \mathcal{L}$ to mean that Δ is a set of such formulae.

Thus, $\mathcal{L}$ combines the language of Computational Tree Logic **CTL** [2] with the language of the full coalitional multi-agent epistemic logic **CMAEL(CD)** [6]. Although $\forall\bigcirc\varphi$ is definable as $\neg\exists\bigcirc\neg\varphi$, it is convenient to treat it as a primitive connective. The operator for individual knowledge $\mathbf{K}_a\varphi$ ("*agent a knows that φ*"), where $a \in \Sigma$, 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 $\exists(\varphi\,\mathcal{U}\psi)$ and $\forall(\varphi\,\mathcal{U}\psi)$ are *temporal eventualities*.

The semantics of temporal-epistemic logics considered in [7] is based on *system of runs* with m processors (agents). A run is a function from (the set of natural numbers) $\mathbb{N}$ to the product L^m regarded as the set of *global states*, where L is the set of *local states*; each agent can be in one of local states at any moment in time. Thus, a global state is a tuple $\langle l_1, \ldots, l_m\rangle$; the i-th component l_i of this global state representing the *local view* of the agent i. The pair (r, n), where r is a run and $n \in \mathbb{N}$, is called in [7] a *point*. With every agent i, the authors of [7] associate the binary *epistemic indistinguishability relation* $\sim_i$ on L^m, defined as follows: $\langle l_1, \ldots, l_m\rangle \sim_i \langle l'_1, \ldots, l'_m\rangle$ if $l_i = l'_i$; i.e., if the agent i has the same local views in these states.

According to [7], a system is *synchronous* when it has a 'global clock' observable by all agents and thus synchronizing their local times; formally, a system is synchronous if $(r, n) \sim_i (r', n')$ implies $n = n'$, for every $i = 1, \ldots, m$, runs r, r', and time moments n, n'. It turns out that the presence or absence of synchrony, under no other assumptions, does not affect the outcome of our tableau procedure, and therefore, the satisfiability of formulae.

The systems with (global) states represented as tuples of local states are generalized in [7] to systems where global states are abstract primitive entities and the epistemic relations are abstract equivalence relations on the set of such states. In the present paper, we work with this abstract semantics from [7]. We note that, as we show later, this semantics is *more general* than the above mentioned 'concrete' semantics from [7], despite the apparent assumption made in [7] that the two semantics are equivalent. We now turn to the presentation of the abstract semantics from [7].

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

1. *Σ is a finite, non-empty set of* agents*;*
2. *$S \neq \emptyset$ is a set of* states*;*

3. $R \neq \emptyset$ *is a set of* runs*: each* $r \in R$ *is a function* $r : \mathbb{N} \mapsto S$*. A state visited by a computation can, then, be represented as* $r(n)$*, where* $r \in R$ *and* $n \in \mathbb{N}$*. Also, with a state* $r(n)$ *we associate a pair* (r, n)*, referred to as a* point*; the set of all points in* $\mathfrak{G}$ *is denoted by* $P(\mathfrak{G})$*. Notice that different points may be associated with the same state.*
4. *for every* $A \in \mathcal{P}^+(\Sigma)$*,* $\mathcal{R}^D_A$ *and* $\mathcal{R}^C_A$ *are binary relations on P(*$\mathfrak{G}$*), such that* $\mathcal{R}^C_A$ *is the reflexive and transitive closure of* $\bigcup_{A' \subseteq A} R^D_{A'}$*.*

A TES $\mathfrak{G}$ *is* synchronous *(STES) if* $((r, n), (r', n')) \in \mathcal{R}^D_A$ *implies* $n = n'$ *for every* $A \in \mathcal{P}^+(\Sigma)$*.*

Hereafter we write '(S)TES' to refer to general or synchronous temporal-epistemic system.

Definition 2.2 *Let* $(r, n) \in P(\mathfrak{G})$ *for some (S)TES* $\mathfrak{G}$ *with a set of runs* R *and let* $r' \in R$*. We say that* r' extends (r, n) *if* $r(m) = r'(m)$ *holds for all* $m \leq n$*.*

Definition 2.3 *A* (synchronous) temporal-epistemic frame *((S)TEF) is a (S)TES* $\mathfrak{G} = (\Sigma, S, R, \{\mathcal{R}^D_A\}_{A \in \mathcal{P}^+(\Sigma)}, \{\mathcal{R}^C_A\}_{A \in \mathcal{P}^+(\Sigma)})$*, where each* $\mathcal{R}^D_A$ *is an equivalence relation satisfying the following condition:*

$$(\dagger) \qquad \mathcal{R}^D_A = \bigcap_{a \in A} \mathcal{R}^D_{\{a\}}$$

If condition (†) is replaced by the following, weaker one:

$$(\dagger\dagger) \quad \mathcal{R}^D_A \subseteq \mathcal{R}^D_B \text{ whenever } B \subseteq A,$$

then $\mathfrak{F}$ *is a* (synchronous) temporal-epistemic pseudo-frame *(pseudo-(S)TEF).*

Notice that in (pseudo-)(S)TEFs $\mathcal{R}^C_A$ is the transitive closure of $\bigcup_{a \in A} R^D_{\{a\}}$, for every $A \in \mathcal{P}^+(\Sigma)$; furthermore, in such structures, each $\mathcal{R}^C_A$ is an equivalence relation.

Definition 2.4 *A* (synchronous) temporal-epistemic model *((S)TEM, for short) is a tuple* $\mathcal{M} = (\mathfrak{F}, L)$*, where*

(*i***)** $\mathfrak{F}$ *is a (S)TEF with a set of runs* R*;*

(*ii***)** $L : R \times \mathbb{N} \mapsto \mathcal{P}(\mathbf{AP})$ *is a* labeling function*, such that* $L(r, n)$ *is the set of atomic propositions 'true' at a point* (r, n)*.*

If condition (i) is changed so that $\mathfrak{F}$ *is a pseudo-(S)TEF, then* $\mathcal{M}$ *is a* (synchronous) temporal-epistemic pseudo-model *(pseudo-(S)TEM).*

Definition 2.5 *The satisfaction relation* $\Vdash$ *between (pseudo-)(S)TEMs, points, and formulae of* $\mathcal{L}$ *is recursively defined as follows:*

$\mathcal{M}, (r, n) \Vdash p$ *iff* $p \in L(r, n)$*;*

$\mathcal{M}, (r, n) \Vdash \neg\varphi$ *iff* $\mathcal{M}, (r, n) \nVdash \varphi$*;*

$\mathcal{M}, (r, n) \Vdash \varphi \wedge \psi$ *iff* $\mathcal{M}, (r, n) \Vdash \varphi$ *and* $\mathcal{M}, (r, n) \Vdash \psi$*;*

$\mathcal{M}, (r, n) \Vdash \exists\bigcirc\varphi$ *iff* $\mathcal{M}, (r', n+1) \Vdash \varphi$ *holds for some* r' *extending* (r, n)*;*

$\mathcal{M}, (r, n) \Vdash \forall\bigcirc\varphi$ *iff* $\mathcal{M}, (r', n+1) \Vdash \varphi$ *holds for every* r' *extending* (r, n)*;*

$\mathcal{M}, (r, n) \Vdash \exists(\varphi\,\mathcal{U}\psi)$ *iff, for some* r' *extending* (r, n)*, there exists* $i \geq n$ *such that* $\mathcal{M}, (r', i) \Vdash \psi$ *and* $\mathcal{M}, (r', j) \Vdash \varphi$ *holds for every* $n \leq j < i$*;*

$\mathcal{M}, (r, n) \Vdash \forall(\varphi\,\mathcal{U}\psi)$ *iff, for every* r' *extending* (r, n)*, there exists* $i \geq n$ *such that* $\mathcal{M}, (r', i) \Vdash \psi$ *and* $\mathcal{M}, (r', j) \Vdash \varphi$ *holds for every* $n \leq j < i$*;*

$\mathcal{M}, (r, n) \Vdash \mathbf{D}_A\varphi$ *iff* $\mathcal{M}, (r', n') \Vdash \varphi$ *for every* $((r, n), (r', n')) \in \mathcal{R}^D_A$*;*

$\mathcal{M}, (r, n) \Vdash \mathbf{C}_A\varphi$ *iff* $\mathcal{M}, (r', n') \Vdash \varphi$ *for every* $((r, n), (r', n')) \in \mathcal{R}^C_A$*.*

Satisfiability and validity of formulae are defined as usual.

Note that in the semantics above *the labeling function* L *acts on points, not states*, i.e., the semantics is *point-based*. To make the semantics *state-based*, one needs to impose the additional condition[1]: $r(n) = r'(n')$ implies $L(r, n) = L(r', n')$. The two semantics differ: e.g., the formula $p \rightarrow \forall(\top\,\mathcal{U}p)$ is valid in the state-based semantics, but not in the point-based one.

The satisfaction condition for the operator $\mathbf{C}_A$ can be paraphrased in terms of reachability. Let $\mathfrak{F}$ be a (pseudo)-(S)TEF over the set of runs R and let $(r, n) \in R \times \mathbb{N}$. We say that point (r', n') is *A-reachable from* (r, n) if either $(r, n) = (r', n')$ 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 $(r_i, n_i), (r_{i+1}, n_{i+1})) \in R^D_{a_i}$. It is then easy to see that the following satisfaction condition for $\mathbf{C}_A$ is equivalent to the one given above:

$\mathcal{M}, (r, n) \Vdash \mathbf{C}_A\varphi$ iff $\mathcal{M}, (r', n') \Vdash \varphi$ for every (r', n'), A-reachable from (r, n).

Note that if $\Sigma = \{a\}$, then $\mathbf{D}_a\varphi \leftrightarrow \mathbf{C}_a\varphi$ is valid in every (S)TEM, for all $\varphi \in \mathcal{L}$. Thus, the single-agent case is essentially trivialized, so we assume throughout the rest of the paper that Σ contains at least 2 (names of) agents.

Also note that in models where states are tuples of local states, if $s \sim_i s'$ holds for every $i = 1, \ldots, m$, then $s = s'$ and, therefore, the formula $p \rightarrow \mathbf{D}_\Sigma p$ is valid in every such model, but it is not valid in every (S)TEM. Thus, the abstract semantics presented above differs from the 'concrete' semantics presented in [7], despite the apparent assumption to the contrary made in [7].

Hereafter, we consider general temporal-epistemic systems; all definitions and results also apply to the synchronous variety, unless stated otherwise.

[1] This condition is not imposed in [7], but this is an apparent omission because it is essentially assumed there.

3 Hintikka structures for CMA-TEL(CD+BT)

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 structure—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 just as much semantic information about θ as is necessary for computing its truth value at a distinguished state. More precisely, while models provide the truth value of every formula of $\mathcal{L}$ at every state, Hintikka structures only determine the truth values of formulae directly involved in the evaluation of a fixed formula θ, in the satisfiability of which 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.

Definition 3.1 *A set $\Delta \subseteq \mathcal{L}$ is* patently inconsistent *if it contains a complementary pair of formulae (i.e., formulae ψ and $\neg\psi$ for some formula ψ).*

A set $\Delta \subseteq \mathcal{L}$ is fully expanded *if it is not patently inconsistent*[2] *and satisfies the following conditions, where $\mathsf{Sub}(\psi)$ stands for the set of subformulae of a formula 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 $\neg\exists\bigcirc\varphi \in \Delta$, then $\forall\bigcirc\neg\varphi \in \Delta$;*
5. *if $\neg\forall\bigcirc\varphi \in \Delta$, then $\exists\bigcirc\neg\varphi \in \Delta$;*
6. *if $\exists(\varphi\,\mathcal{U}\psi) \in \Delta$, then $\psi \in \Delta$ or $\varphi, \exists\bigcirc\exists(\varphi\,\mathcal{U}\psi) \in \Delta$;*
7. *if $\neg\exists(\varphi\,\mathcal{U}\psi) \in \Delta$, then $\neg\psi, \neg\varphi \in \Delta$ or $\neg\psi, \neg\exists\bigcirc\exists(\varphi\,\mathcal{U}\psi) \in \Delta$;*
8. *if $\forall(\varphi\,\mathcal{U}\psi) \in \Delta$, then $\psi \in \Delta$ or $\varphi, \forall\bigcirc\forall(\varphi\,\mathcal{U}\psi) \in \Delta$;*
9. *if $\neg\forall(\varphi\,\mathcal{U}\psi) \in \Delta$, then $\neg\psi, \neg\varphi \in \Delta$ or $\neg\psi, \neg\forall\bigcirc\forall(\varphi\,\mathcal{U}\psi) \in \Delta$;*
10. *if $\mathbf{D}_A\varphi \in \Delta$, then $\mathbf{D}_{A'}\varphi \in \Delta$ for every A' such that $A \subseteq A' \subseteq \Sigma$;*
11. *if $\mathbf{D}_A\varphi \in \Delta$, then $\varphi \in \Delta$;*
12. *if $\mathbf{C}_A\varphi \in \Delta$, then $\mathbf{D}_a(\varphi \wedge \mathbf{C}_A\varphi) \in \Delta$ for every $a \in A$;*
13. *if $\neg\mathbf{C}_A\varphi \in \Delta$, then $\neg\mathbf{D}_a(\varphi \wedge \mathbf{C}_A\varphi) \in \Delta$ for some $a \in A$;*
14. *if $\psi \in \Delta$ and $\mathbf{D}_A\varphi \in \mathsf{Sub}(\psi)$, then either $\mathbf{D}_A\varphi \in \Delta$ or $\neg\mathbf{D}_A\varphi \in \Delta$.*

[2] Even though in general, not being patently inconsistent is a weaker condition than a propositional consistency, in the case of fully expanded sets, they coincide.

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 $(r, n) \in R \times \mathbb{N}$ with sets of formulae of $\mathcal{L}$ satisfying the following constraints, for all (r, n):*

H1 *$H(r, n)$ is fully expanded;*
H2 *if $\neg\varphi \in H(r, n)$, then $\varphi \notin H(r, n)$;*
H3 *if $\exists\bigcirc\varphi \in H(r, n)$, then $\varphi \in H(r', n + 1)$ holds for some r' extending (r, n);*
H4 *if $\forall\bigcirc\varphi \in H(r, n)$, then $\varphi \in H(r', n + 1)$ holds for every r' extending (r, n);*
H5 *if $\exists(\varphi\,\mathcal{U}\psi) \in H(r, n)$, then, for some r' extending (r, n), there exists $i \geq n$ such that $\psi \in H(r', i)$ and $\varphi \in H(r', j)$ holds for every $n \leq j < i$;*
H6 *if $\forall(\varphi\,\mathcal{U}\psi) \in H(r, n)$, then, for every r' extending (r, n), there exists $i \geq n$ such that $\psi \in H(r', i)$ and $\varphi \in H(r', j)$ holds for every $n \leq j < i$;*
H7 *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 $\neg\varphi \in H(r', n')$;*
H8 *if $((r, n), (r', n')) \in \mathcal{R}_A^D$, then $\mathbf{D}_{A'}\varphi \in H(r, n)$ iff $\mathbf{D}_{A'}\varphi \in H(r', n')$ holds for every $A' \subseteq A$;*
H9 *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 temporal-epistemic Hintikka structures are defined accordingly.

Definition 3.3 *A formula θ is* satisfiable *in a TEHS $\mathcal{H}$ with a labeling function H if $\theta \in H(r, n)$ for some point (r, n) of $\mathcal{H}$. A set of formulae Θ is* satisfiable *in $\mathcal{H}$ if $\Theta \subseteq H(r, n)$ for some point (r, n) of $\mathcal{H}$.*

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 $\mathcal{M}$, we define the *extended labeling* $L^+_{\mathcal{M}}$ on the set of points of $\mathcal{M}$ as follows: $L^+_{\mathcal{M}}(r, n) = \{\,\varphi \mid \mathcal{M}, (r, n) \Vdash \varphi\,\}$ for every (r, n). The following claim is then straightforward.

Lemma 3.4 *Let $\mathcal{M} = (\Sigma, S, R, \{\mathcal{R}_A^D\}_{A\in\mathcal{P}^+(\Sigma)}, \{\mathcal{R}_A^C\}_{A\in\mathcal{P}^+(\Sigma)}, L)$ be a TEM satisfying $\theta \in \mathcal{L}$, and let $L^+_{\mathcal{M}}$ be the extended labeling on $\mathcal{M}$. Then, $\mathcal{H} = (\Sigma, S, R, \{\mathcal{R}_A^D\}_{A\in\mathcal{P}^+(\Sigma)}, \{\mathcal{R}_A^C\}_{A\in\mathcal{P}^+(\Sigma)}, L^+)$ is a TEHS satisfying θ.*

To establish the converse, we first prove that the existence of a Hintikka structure satisfying θ implies the existence of a pseudo-model satisfying θ; then, we prove that this in turn implies the existence of a model satisfying θ.

Lemma 3.5 *If $\theta \in \mathcal{L}$ is satisfiable in a TEHS, then it is satisfiable in a pseudo-TEM.*

Proof. Let $\mathcal{H} = (\Sigma, S, R, \{\mathcal{R}^D_A\}_{A\in\mathcal{P}^+(\Sigma)}, \{\mathcal{R}^C_A\}_{A\in\mathcal{P}^+(\Sigma)}, H)$ be a TEHS satisfying θ. We build a pseudo-TEM satisfying θ as follows. First, for every $A \in \mathcal{P}^+(\Sigma)$, let $\mathcal{R}'^D_A$ be the reflexive, symmetric, and transitive closure of $\bigcup_{A\subseteq B}\mathcal{R}^D_B$ and let $\mathcal{R}'^C_A$ be the transitive closure of $\bigcup_{a\in A}\mathcal{R}'^D_a$. Notice that $\mathcal{R}^D_A \subseteq \mathcal{R}'^D_A$ and $\mathcal{R}^C_A \subseteq \mathcal{R}'^C_A$ for every $A \in \mathcal{P}^+(\Sigma)$. Next, let $L(r,n) = H(r,n) \cap \mathbf{AP}$, for every point $(r,n) \in R \times \mathbb{N}$. It is then easy to check that $\mathcal{M}' = (\Sigma, S, R, \{\mathcal{R}'^D_A\}_{A\in\mathcal{P}^+(\Sigma)}, \{\mathcal{R}'^C_A\}_{A\in\mathcal{P}^+(\Sigma)}, L)$ is a pseudo-TEM.

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 $p \in \mathbf{AP}$. 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 (H2), $p \notin H(r,n)$ 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 $\neg\neg\varphi \in H(r,n)$, then by virtue of (H1), $\varphi \in H(r,n)$ and hence, by inductive hypothesis, $\mathcal{M}', (r,n) \Vdash \varphi$ and thus $\mathcal{M}', (r,n) \Vdash \neg\neg\varphi$.

The cases of $\chi = \varphi \wedge \psi$, $\chi = \exists\bigcirc\varphi$, and $\chi = \forall\bigcirc\varphi$ are straightforward, using (H1) – (H4).

Let χ be $\exists(\varphi\,\mathcal{U}\psi)$. If $\exists(\varphi\,\mathcal{U}\psi) \in H(r,n)$, then the desired conclusion immediately follows from (H5) and the inductive hypothesis. If $\neg\exists(\varphi\,\mathcal{U}\psi) \in H(r,n)$, then due to (H1), either $\neg\psi, \neg\varphi \in H(r,n)$ or $\neg\psi, \neg\exists\bigcirc\exists(\varphi\,\mathcal{U}\psi) \in H(r,n)$. In the former case, the conclusion immediately follows from the inductive hypothesis. Otherwise, due to (H1) and (H4), $\neg\exists(\varphi\,\mathcal{U}\psi) \in H(r', n+1)$ holds for every run r' extending (r,n). By repeating the argument, we obtain that, for every run r' extending (r,n), either $\neg\varphi \in H(r',i)$ for some $i \geq 0$ and $\neg\psi \in H(r',j)$ for every $0 \leq j \leq i$ or $\neg\psi \in H(r',i)$ for every $i \geq 0$. In either case, the inductive hypothesis implies that $\mathcal{M}, (r,n) \Vdash \neg\exists(\varphi\,\mathcal{U}\psi)$, as desired.

The case of $\chi = \forall(\varphi\,\mathcal{U}\psi)$ is similar to the previous one and is left to the reader.

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 $((r,n),(r',n')) \in \mathcal{R}'^D_A$ implies $\varphi \in H(r',n')$. So, assume that $((r,n),(r',n')) \in \mathcal{R}'^D_A$. There are two cases to consider. If $(r,n) = (r',n')$, then the conclusion immediately follows from (H1). Otherwise, there exists an undirected path from (r,n) to (r',n') along the relations $\mathcal{R}^D_{A'}$, where each A' is a superset of A. Then, due to (H8), $\mathbf{D}_A\varphi \in H(r',n')$; hence, by (H1), $\varphi \in H(r',n')$, as desired.

Now, let $\neg\mathbf{D}_A\varphi \in H(r,n)$. In view of the inductive hypothesis, it suffices to show that there exist $r' \in R$ and $n' \in \mathbb{N}$ such that $((r,n),(r',n')) \in \mathcal{R}'^D_A$ and $\neg\varphi \in H(r',n')$. By (H7), there exists $r' \in R$ and $n' \in \mathbb{N}$ such that $((r,n),(r',n')) \in \mathcal{R}^D_A$ and $\neg\varphi \in H(r',n')$. As $\mathcal{R}^D_A \subseteq \mathcal{R}'^D_A$, the desired conclusion follows.

Suppose that $\chi = \mathbf{C}_A\varphi$. Assume that $\mathbf{C}_A\varphi \in H(r,n)$. In view of the inductive hypothesis, it suffices to show that if (r',n') is A-reachable from (r,n) in $\mathcal{M}'$, then $\varphi \in H(r',n')$. If $(r,n) = (r',n')$ the claim follows from (H1). So, suppose that, 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 $((r_i,n_i),(r_{i+1},n_{i+1})) \in \mathcal{R}'^D_{a_i}$. Then, for every $0 \leq i < m$, there exists a path from (r_i,n_i) to (r_{i+1},n_{i+1}) along relations $\mathcal{R}^D_{A'}$ such that $a_i \in A'$ for every A'. Then, we can show by induction on i, using (H1) and (H8), that $\mathbf{C}_A\varphi \in H(r_i,n_i)$ holds for every $0 \leq i < m$. Indeed, this holds for $i = 0$; assuming that it holds for some i, by (H1)(12) we have that $\mathbf{D}_{a_i}(\varphi \wedge \mathbf{C}_A\varphi) \in H(r_i,n_i)$, hence, by (H1)(10) and (H8), $\varphi \in H(r_{i+1},n_{i+1})$. Now, by taking $i = m-1$ we obtain $\varphi \in H(r',n')$, as required.

Finally, assume that $\neg\mathbf{C}_A\varphi \in H(r,n)$. Then, the desired conclusion follows from (H9), the fact that $\mathcal{R}^C_A \subseteq \mathcal{R}'^C_A$, and the inductive hypothesis. □

Lemma 3.6 *If $\theta \in \mathcal{L}$ is satisfiable in a pseudo-TEM, then it is satisfiable in a TEM.*

Proof. The proof is exactly the same as in [5, Section 3], as the pseudo-models are only 'defective' with respect to epistemic, but not temporal, relations; therefore, the construction for branching time is the same as for linear time. □

Lemmas 3.4, 3.5, and 3.6 immediately give us the following theorem.

Theorem 3.7 *A $\theta \in \mathcal{L}$ is satisfiable in a TEM iff it is satisfiable in a TEHS.*

4 Tableau procedure for CMATEL(CD+BT)

In this section, we present a tableau procedure for **CMATEL(CD+BT)**. We describe a procedure for testing for satisfiability in synchronous models, as it requires extra care. We then briefly mention how the general case is different

and argue that the outcome of the procedure is the same in both cases, implying that, satisfiability-wise, general and synchronous semantics are equivalent.

4.1 Overview of the 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 sets of $\mathcal{L}$-formulae, encoding 'sufficiently many' TEHSs for θ, in the sense that if θ is satisfiable, then it is satisfiable in a TEHS represented by $\mathcal{T}^\theta$. The main ideas underlying our tableau algorithm come from the tableau procedures for the logics **PDL** in [9], **UB** in [1] and **CTL** in [3] (see also a detailed exposition of tableaux for **CTL** in [10]), as well as recently developed tableaux for multiagent epistemic logics in [6]. To make the present paper self-contained, we outline the basic ideas behind our tableau algorithm in line with those references, and then describe the particulars specific to **CMATEL(CD+BT)**.

Usually, tableaux work by decomposing the input formula into simpler formulae, in accordance with the semantics of the logical connectives. 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 propositional analogue of Hintikka structures, for the input formula θ. If at least one leaf of that tree is a Hintikka set for θ, the search has succeeded and θ is proved satisfiable; otherwise, it is declared unsatisfiable.

When applied to logics containing fixpoint-definable operators, such as $\mathbf{C}_A$, $\exists\mathcal{U}$, and $\forall\mathcal{U}$, these two defining features of the classical tableau method no longer apply. First, the decomposition of the fixpoint formulae, which is done by unfolding their fixpoint definitions, usually produces larger formulae: $\mathbf{C}_A\varphi$ is decomposed into the formulae $\mathbf{D}_a(\varphi \wedge \mathbf{C}_A\varphi)$; analogously for formulae of the form $\exists(\varphi\,\mathcal{U}\psi)$ and $\forall(\varphi\,\mathcal{U}\psi)$. Hence, we need a termination-ensuring mechanism. In our tableaux, such a mechanism is provided by the use (and reuse) of so called "pre-states", 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+BT)**, there are other such reasons, specific to TEHS, which are more involved structures than classical Hintikka sets. One such reason has to do with eventualities: the truth of an eventuality at a state s in a TEM $\mathcal{M}$ requires existence of a path going from s to a state of $\mathcal{M}$ at which the 'promise' of that eventuality is fulfilled. Since truth in TEMs is simulated by membership in state labels of Hintikka structures, eventualities impose respective conditions on the labels. Thus, 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 (i.e. a path along relations of the form $\mathcal{R}_B^D$, where $\mathcal{R}_B^D \subseteq \mathcal{R}_A^D$) from s to a state t whose label contains $\neg\varphi$, due to condition (H9) of Definition 3.2. Similar requirements apply to eventualities of the form $\exists(\varphi\,\mathcal{U}\psi)$ and $\forall(\varphi\,\mathcal{U}\psi)$ due to conditions (H5) and (H6) of Definition 3.2. The tableau analogs of these conditions are called *realization of eventualities*. If a tableau contains a node with an unrealized eventuality in its label, then it cannot produce a TEHS, and thus is 'bad' and needs repairing by removing such nodes. The third possible reason for a tableau to be 'bad' has to do with successor nodes: it may so happen that some of the required successors of a node s are missing from the tableau; then, s is 'bad', and hence needs to be removed. Notice that in TEHSs, and thus in tableaux, states have two kinds of successors: temporal and epistemic. The absence 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*, *pre-state elimination*, and *state elimination*. During the first, we produce the *pretableau* for θ—a directed graph $\mathcal{P}^\theta$, from which the tableau $\mathcal{T}^\theta$ will be extracted. The nodes of $\mathcal{P}^\theta$ are sets of formulae coming in two varieties: *states* and *pre-states*. States are fully expanded sets, meant to represent (labels of) states of a Hintikka structure, while pre-states only play a temporary role in the construction of $\mathcal{T}^\theta$. During the second phase, all pre-states from $\mathcal{P}^\theta$ are removed and their incoming edges are redirected, creating a smaller graph $\mathcal{T}_0^\theta$, the *initial tableau for* θ. Finally, we remove from $\mathcal{T}_0^\theta$ all states, if any, that cannot be satisfied in a TEHS, for any of the reasons mentioned above. 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. An example illustrating the tableau construction is provided in Appendix A.

4.2 Pretableau construction phase

All states and pre-states of the pretableau $\mathcal{P}^\theta$ constructed during this phase are 'time-stamped'. Whenever necessary to make it explicit, we will use the notation $\Gamma^{[k]}$ indicating that pre-state Γ was created as the kth component of a run; likewise for states.

The pretableau contains three types of edge, described below. As already mentioned, a procedure attempts to produce a compact representation of a sufficiently many TEHSs for the input formula θ by organizing an exhaustive search for such structures. One type of edge, depicted by

unmarked double arrows $\Longrightarrow$, represents the search transitions in the tableau. The exhaustive search considers all possible alternatives that arise when expanding pre-states into states through branching when dealing with disjunctive formulae. Thus, when we draw a double arrow from a pre-state 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 pre-states.

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 pre-state $\Gamma^{[k]}$ such that **(SR)** has not been applied to $\Gamma^{[k]}$ earlier, do the following:

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

We denote by $\mathbf{states}(\Gamma)$ the (finite) set of states $\{\, \Delta \mid \Gamma \Longrightarrow \Delta \,\}$.

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 (pre)state being part of two different runs at different moments of time; 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 TEHS 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 $\neg\mathbf{D}_A\varphi \in \Delta^{[k]}$, then we need some pre-state $\Gamma^{[k]}$ containing $\neg\varphi$ to be accessible from $\Delta^{[k]}$ by $\mathcal{R}^D_A$.[3] The reason we mark these single arrows by a formula $\neg\mathbf{D}_A\varphi$ (rather than by just coalition A), is that we have to remember why we had to create this particular Γ, and not just what relation connects Δ to Γ. This information will be needed during the elimination phases. We now formulate the rule producing this second type of edge.

[3]We require the newly created pre-states to bear the same time stamp as the source state for the sake of synchrony, as this reflects the fact that all epistemic alternatives belong to the same temporal level of any TEHS.

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

1. Create a new pre-state $\Gamma^{[k]} = \{\neg\varphi\} \cup \bigcup_{A'\subseteq A}\{\, \mathbf{D}_{A'}\psi \mid \mathbf{D}_{A'}\psi \in \Delta \,\} \cup \bigcup_{A'\subseteq A}\{\, \neg\mathbf{D}_{A'}\psi \mid \neg\mathbf{D}_{A'}\psi \in \Delta \,\}$.
2. If pre-state $\Gamma^{[k]}$ is patently inconsistent, remove it.
3. Otherwise, connect $\Delta^{[k]}$ to $\Gamma^{[k]}$ with $\stackrel{\neg\mathbf{D}_A\varphi}{\longrightarrow}$.
4. If, however, the tableau already contains a pre-state $\Gamma'^{[k]} = \Gamma^{[k]}$, do not add another copy of $\Gamma'^{[k]}$, but simply connect $\Delta^{[k]}$ to $\Gamma'^{[k]}$ with $\stackrel{\neg\mathbf{D}_A\varphi}{\longrightarrow}$.

The third type of edge, depicted by single arrows marked with formulae of the form $\exists\bigcirc\varphi$, represent temporal transitions in TEHSs that the tableau is trying to build. The rationale for this rule is similar to that for **(DR)**, the only difference being that we are now considering temporal, rather than epistemic, formulae forcing creation of new pre-states.

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

1. For each $\exists\bigcirc\varphi \in \Delta^{[k]}$, create a new pre-state $\Gamma^{[k+1]} = \{\varphi\} \cup \{\, \psi \mid \forall\bigcirc\psi \in \Delta^{[k]} \,\}$.[4]
2. If pre-state $\Gamma^{[k]}$ is patently inconsistent, remove it immediately.
3. Otherwise, connect $\Delta^{[k]}$ to $\Gamma^{[k+1]}$ with $\stackrel{\exists\bigcirc\varphi}{\longrightarrow}$.
4. If, however, the tableau already contains a pre-state $\Gamma'^{[m]} = \Gamma^{[k+1]}$, do not add another copy of $\Gamma'^{[m]}$, but simply connect $\Delta^{[k]}$ to $\Gamma'^{[m]}$ with $\stackrel{\exists\bigcirc\varphi}{\longrightarrow}$.

We now describe the order of application of the above rules. We start off by creating a single pre-state $\{\theta\}$, containing the input formula. Then, we alternatingly apply **(DR)** and **(Next)** to the states created at the previous stage and then applying **(SR)** to the newly created pre-states. The construction stage is over when the applications of **(DR)** and **(Next)** do not produce any new pre-states.

4.3 Pre-State elimination phase

At this phase we remove from $\mathcal{P}^\theta$ all pre-states and double arrows, which results in a smaller graph $\mathcal{T}^\theta_0$ called the *initial tableau*. Formally, we apply the following rule:

Rule (PR) For every pre-state 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 $\Delta \stackrel{\chi}{\longrightarrow} \Gamma$, then for every state $\Delta' \in \mathbf{states}(\Gamma)$, put $\Delta \stackrel{\chi}{\longrightarrow} \Delta'$.

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

4.4 State elimination phase

During this phase we remove from $\mathcal{T}_0^\theta$ states that are not satisfiable in a TEHS. As we do not create patently inconsistent states, 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 turn out unsatisfiable, or Δ contains an eventuality that is not realized in the tableau. Accordingly, we have two elimination rules: **(E1)** 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 exactly one state from the tableau $\mathcal{T}_n^\theta$ obtained at the previous stage, by applying one of the elimination rules, obtaining the tableau $\mathcal{T}_{n+1}^\theta$. In the rules below, S_m^θ denotes the set of states of tableau $\mathcal{T}_m^\theta$.

(E1) If Δ contains a formula χ of the form $\neg\mathbf{D}_A\varphi$ or $\exists\bigcirc\varphi$, 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 $\mathcal{T}_n^\theta$.

For the other elimination rule, we need the concept of *eventuality realization* in a tableau.

Definition 4.1 (Eventuality realization)

- *The eventuality $\neg\mathbf{C}_A\varphi$ is* realized at Δ in $\mathcal{T}_n^\theta$ *if there exists a 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 $\Delta_i \xrightarrow{\chi_i} \Delta_{i+1}$.*
- *The eventuality $\exists(\varphi\,\mathcal{U}\psi)$* is realized at Δ in $\mathcal{T}_n^\theta$ *if there exists a path $\Delta = \Delta_0, \Delta_1, \ldots, \Delta_m$, where $m \geq 0$, such that $\psi \in \Delta_m$, and for every $0 \leq i < m$, there exist a formula χ_i such that $\Delta_i \xrightarrow{\exists\bigcirc\chi_i} \Delta_{i+1}$ and $\varphi \in \Delta_i$.*
- *For eventualities of the form $\forall(\varphi\,\mathcal{U}\psi)$, we define the notion* "is realized at Δ in $\mathcal{T}_n^\theta$" *recursively as follows:*

  (i) If $\psi \in \Delta$ then $\forall(\varphi\,\mathcal{U}\psi)$ is realized at Δ;

  (ii) If $\varphi \in \Delta$ and, for every $\exists\bigcirc\chi \in \Delta$, there is a state $\Delta' \in \mathcal{T}_n^\theta$ such that $\Delta \xrightarrow{\exists\bigcirc\chi} \Delta'$ and $\forall(\varphi\,\mathcal{U}\psi)$ is realized at Δ', then $\forall(\varphi\,\mathcal{U}\psi)$ is realized at Δ.

Now, we can state our second state elimination rule.

(E2) If $\Delta \in S_n^\theta$ contains an 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 procedures that eventually marks all states realizing a given eventuality ξ in $\mathcal{T}_n^\theta$:

- If $\xi = \neg\mathbf{C}_A\varphi$, then we initially mark all $\Delta \in S_n^\theta$ such that $\neg\varphi \in \Delta$. Then, we repeat the following subprocedure until no more states get marked: for every still unmarked $\Delta \in S_n^\theta$, mark Δ if there is at least one marked Δ' such that $\Delta \xrightarrow{\mathbf{D}_B\psi} \Delta'$, for some $B \subseteq A$.
- If $\xi = \exists(\varphi\,\mathcal{U}\psi)$, then we initially mark all $\Delta \in S_n^\theta$ such that $\psi \in \Delta$. Then, we repeat the following subprocedure until no more states get marked: for every still unmarked $\Delta \in S_n^\theta$, mark Δ if $\varphi \in \Delta$ and there is at least one marked Δ' such that $\Delta \xrightarrow{\exists\bigcirc\xi} \Delta'$.
- If $\xi = \forall(\varphi\,\mathcal{U}\psi)$, then we initially mark all $\Delta \in S_n^\theta$ such that $\psi \in \Delta$, and then we repeat the following subprocedure until no more states get marked: for every still unmarked $\Delta \in S_n^\theta$, mark Δ if $\varphi \in \Delta$ and, for every formula $\exists\bigcirc\chi \in \Delta$, there is a marked state $\Delta' \in S_n^\theta$ such that $\Delta \xrightarrow{\exists\bigcirc\chi} \Delta'$.

We now describe the order of application of the above rules. We have to be careful, since having applied **(E2)** to a tableau, we could have removed all the states accessible from some Δ along the arrows marked with some χ; hence, we need to reapply **(E1)** to the resultant tableau to remove such Δ's. Conversely, having applied **(E1)**, we could have thrown away states needed for realizing certain eventualities; hence, we need to reapply **(E2)**. Thus, we need to apply **(E1)** and **(E2)** in an alternating sequence that cycles through all eventualities. More precisely, we arrange all eventualities occurring in the states of $\mathcal{T}_0^\theta$ 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 **(E1)** to the resulting tableau, until all the eventualities have been dealt with, i.e., we have 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 $\mathcal{T}^\theta$, whose set of states is denoted by S^θ.

Definition 4.2 *The final tableau $\mathcal{T}^\theta$ is* open *if $\theta \in \Delta$ for some $\Delta \in S^\theta$; otherwise, $\mathcal{T}^\theta$ is* closed.

If the final tableau is closed, the tableau procedure returns "no"; otherwise, it returns "yes".

We briefly mention that, to test for satisfiability in general models, we relax the rule **(DR)**, allowing states to have epistemic successors from different temporal levels. As such a modification does not result in the outcome of the procedure, we conclude that, satisfiability-wise, the semantics based on general models is equivalent to the one based on synchronous models.

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. We give the proofs for the synchronous case, the modification for the general case being straightforward. The proofs of the following three lemmas are straightforward.

Lemma 5.1 *Let Γ be a pre-state 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 $\Delta \in S^\theta$ 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 $\Delta \in S^\theta$ be such that $\mathcal{M}, (r,n) \Vdash \Delta$ for some TEM $\mathcal{M}$ and point (r,n), and $\exists\bigcirc\varphi \in \Delta$. Then, $\mathcal{M}, (r',n+1) \Vdash \{\varphi\} \cup \{\, \psi \mid \forall\bigcirc\psi \in \Delta \,\}$ holds for some r' extending (r,n).*

Lemma 5.4 *Let $\Delta \in S^\theta$ 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 $\mathcal{T}^\theta$.*

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 $\neg\varphi$. As the tableau organizes an exhaustive search, a chain of tableau states corresponding to those states in the model will be produced. □

The next two lemmas are proved likewise.

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

Lemma 5.6 *Let $\Delta \in S^\theta$ be such that $\mathcal{M}, (r,n) \Vdash \Delta$ for some TEM $\mathcal{M}$ and a point (r,n), and let $\forall(\varphi\,\mathcal{U}\psi) \in \Delta$. Then, $\forall(\varphi\,\mathcal{U}\psi)$ is realized at Δ in $\mathcal{T}^\theta$.*

Theorem 5.7 *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 **(E1)** or **(E2)**. 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 3.7, it suffices to show that an open tableau for θ can produce a TEHS satisfying θ. Moreover, we show that this TEHS can be constructed synchronous.

Lemma 5.8 *If $\mathcal{T}^\theta$ is open, then θ is satisfiable in a (synchronous) TEHS.*

Proof sketch. We build the TEHS $\mathcal{H}$ for θ by induction on the temporal levels. The main concern is to ensure that all eventualities in the resultant structure are realized, i.e. the properties (H5), (H6) and (H9) from the definition of Hintikka structures hold; all the other properties of Hintikka structures transfer, more or less immediately, 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 $\exists(\varphi\,\mathcal{U}\psi)$ and $\forall(\varphi\,\mathcal{U}\psi)$). Essentially, the construction combines the construction used in proving completeness of multi-agent epistemic Hintikka structures from [6] and the one used in proving completeness of **CTL** (see [10], which essentially uses the construction that is a simplification of the construction for **ATL** from [4]).

We start by building the 0th level of our prospective Hintikka structure from the level 0 of an open tableau. For each state $\Delta^{[0]}$ on this level, if $\Delta^{[0]}$ does not contain any epistemic eventualities, we define $\Delta^{[0]}$-epistemic component 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 the $\Delta^{[0]}$-epistemic component is a tree obtained from a path in the tableau leading from $\Delta^{[0]}$ along the arrows marked with formulae of the form $\neg\mathbf{D}_B\chi$ to a state $\Delta'^{[0]}$ containing φ; the tree is obtained from the path by giving each component of the path enough successors, as described above. 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 graph where each 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 **CTL** ([10]): we define the $\Delta^{[0]}$-temporal component for each $\Delta^{[0]}$ as follows: if it does not contain any temporal eventualities, then we take $\Delta^{[0]}$ with one temporal successor for each $\exists\bigcirc\varphi \in \Delta$. If $\exists(\varphi\,\mathcal{U}\psi) \in \Delta$, then we take a temporal path realizing $\exists(\varphi\,\mathcal{U}\psi) \in \Delta$ and give to every node enough temporal successors, as describe above. Lastly, if $\forall(\varphi\,\mathcal{U}\psi) \in \Delta$, then we take a temporal tree witnessing the the realizion of $\forall\varphi \in \Delta$ in the tableau (for details, see [10]). As eventualities are again passed down, we can stitch up an infinite tree realizing all the eventualities contained in the states making up the tree.

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 the 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). 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 $H(\Delta^{[k]}) = \Delta^{[k]}$ for every $\Delta^{[k]}$, to obtain a TEHS satisfying θ. □

The completeness is now immediate from Lemma 5.8 and Theorem 3.7.

Theorem 5.9 (Completeness) *If $\mathcal{T}^\theta$ is open, then θ is satisfiable, in a (synchronous) TEM.*

As for the complexity, for lack of space, we only mention that the above procedure runs in exponential time (the calculations are fairly routine), thus matching the lower bound known from [7].

6 Concluding remarks

We have presented an incremental tableau-based decision procedure for the full coalitional temporal-epistemic logic of branching time **CMATEL(CD+BT)**. The procedure is complexity-optimal, intuitive, and practically reasonably efficient (as the number of (pre)states it creates is usually significantly smaller that the powerset of all subsets of the close of the formula that is tested for satisfiability); it is, therefore, suitable for both manual and automated execution. Moreover, it is fairly flexible and easily amenable to modifications for variations of the semantics, such as those mentioned in section 2. Since in the semantics considered in this paper there is essentially no interaction between the temporal and epistemic fragments, our procedure combines in a modular way tableaux for the full coalitional multiagent epistemic logic **CMAEL(CD)** and for **CTL**. Such interaction, however, can be triggered by imposing various natural semantic conditions, such as "no learning" or "no forgetting". As shown in [7], such conditions may increase dramatically the complexity of the logic, up to highly undecidable. However, even for the relatively 'easy' cases of EXPSPACE-hard logics, the construction of a tableau procedures is still an open challenge, which we intend to address in the future.

References

[1] Mordechai Ben-Ari, Amir Pnueli, and Zohar Manna. The temporal logic of branching time. *Acta Informatica*, 20:207–226, 1983.

[2] E. Allen Emerson. Temporal and modal logics. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, pages 995–1072. MIT Press, 1990.

[3] E. Allen Emerson and Joseph Halpern. Decision procedures and expressiveness in the temporal logic of branching time. *Journal of Computer and System Sciences*, 30(1):1–24, 1985.

[4] Valentin Goranko and Dmitry Shkatov. Tableau-based decision procedures for logics of strategic ability in multi-agent systems. To appear in *ACM Transactions on Computational Logic*. Available at `http://tocl.acm.org/accepted.html`.

[5] Valentin Goranko and Dmitry Shkatov. Tableau-based decision procedure for full coalitional multiagent temporal-epistemic logic of linear time. In Decker, Sichman, Sierra, and Castelfranchi, editors, *Proc. of the 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009), May 2009, Budapest, Hungary*, 2009.

[6] Valentin Goranko and Dmitry Shkatov. Tableau-based procedure for deciding satisfiability in the full coalitional multiagent epistemic logic. In Sergei Artemov and Anil 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.

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

[8] Agi Kurucz, Frank Wolter, Michael Zakharyaschev, and Dov Gabbay. *Multi-dimentional Modal Logics: Theory and Applications*. Elsevier, 2003.

[9] Vaughan R. Pratt. A practical decision method for propositional dynamic logic. In *Proceedings of the 10th Annual ACM Symposium on the Theory of Computing*, pages 326–227, San Diego, California, May 1979.

[10] Dmitry Shkatov. Incremental **CTL**-tableaux revisited. Submitted.

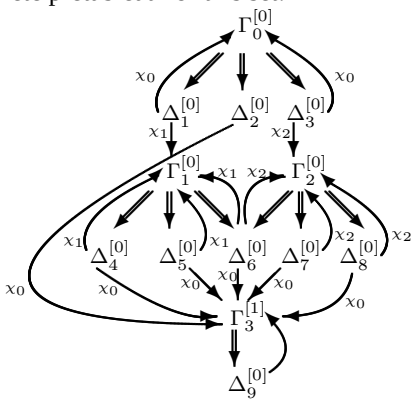
[11] Dirk Walther. ATEL with common and distributed knowledge is ExpTime-complete. In *Proceedings of Methods for Modalities 4*, Berlin, 2005.

[12] Dirk Walther, Carsten Lutz, Frank Wolter, and Michael Wooldridge. ATL satisfiability is indeed ExpTime-complete. *Journal of Logic and Computation*, 16(6):765–787, 2006.

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

A Example

In the present appendix, we provide an example of how our procedure works on the formula $\forall(\neg\mathbf{C}_{\{a,b\}}p\,\mathcal{U}\,\neg\mathbf{D}_{\{a,c\}}p)$. To simplify the example, we test for satisfiability of the equivalent set $\{\forall(\neg\mathbf{C}_{\{a,b\}}p\,\mathcal{U}\,\neg\mathbf{D}_{\{a,c\}}p, \top\}$. Displayed below is the complete pretableau for this set.



$\chi_0 = \exists\bigcirc\top$;
$\chi_1 = \neg\mathbf{D}_a(p \wedge \mathbf{C}_{\{a,b\}}p)$;
$\chi_2 = \neg\mathbf{D}_b(p \wedge \mathbf{C}_{\{a,b\}}p)$;
$\Gamma_0 = \{\forall(\neg\mathbf{C}_{\{a,b\}}p\,\mathcal{U}\,\mathbf{D}_{\{a,c\}}p) = \theta, \top\}$;
$\Delta_1 = \{\theta, \neg\mathbf{C}_{\{a,b\}}p, \forall\bigcirc\theta, \chi_1, \exists\bigcirc\top\}$;
$\Delta_2 = \{\mathbf{D}_{\{a,c\}}p, p, \exists\bigcirc\top\}$;
$\Delta_3 = \{\theta, \neg\mathbf{C}_{\{a,b\}}p, \forall\bigcirc\theta, \chi_2, \exists\bigcirc\top\}$;
$\Gamma_1 = \{\chi_1, \neg(p \wedge \mathbf{C}_{\{a,c\}}p)\}$;
$\Gamma_2 = \{\chi_2, \neg(p \wedge \mathbf{C}_{\{a,c\}}p)\}$;
$\Delta_4 = \{\chi_1, \neg p, \exists\bigcirc\top\}$;
$\Delta_5 = \{\chi_1, \neg\mathbf{C}_{\{a,b\}}p, \exists\bigcirc\top\}$;
$\Delta_6 = \{\chi_1, \neg\mathbf{C}_{\{a,b\}}p, \chi_2, \exists\bigcirc\top\}$;
$\Delta_7 = \{\chi_2, \neg\mathbf{C}_{\{a,c\}}p, \exists\bigcirc\top\}$;
$\Delta_8 = \{\chi_2, \neg p, \exists\bigcirc\top\}$;
$\Gamma_3 = \{\top\}$;
$\Delta_9 = \{\top, \exists\bigcirc\top\}$.

The initial tableau is obtained by removing all pre-states (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.