Clausal Graph Tableaux for Hybrid Logic with Eventualities and Difference

Mark Kaminski and Gert Smolka
Saarland University

August 10, 2010

We introduce the method of clausal graph tableaux at the example of hybrid logic with difference and star modalities. Clausal graph tableaux are prefix-free and terminate by construction. They provide an abstract method of establishing the small model property of modal logics. In contrast to the filtration method, clausal graph tableaux result in goal-directed decision procedures. Until now no goal-directed decision procedure for the logic considered in this paper was known. There is the promise that clausal graph tableaux lead to a new class of effective decision procedures.

1 Introduction

For modal logic there exist two basic kinds of tableau decision procedures. The more traditional kind, dating back to Kripke [20] and further developed by Fitting [9], Massacci [23] and others, sees tableaux as rooted trees labeled with formulas and, sometimes, auxiliary meta-level information. The formulas on an individual tableau branch are interpreted conjunctively while the different branches are interpreted disjunctively. Tableau calculi are designed so that the set of all branches of a tableau represents an exhaustive enumeration of ways to satisfy the formula at the root of the tableau. A typical decision procedure for satisfiability based on tree tableaux will explore the tableau branch by branch until it finds an evident branch. An evident branch is a satisfiable branch that syntactically describes a model of all of its formulas, in particular of the root formula. Most modern tree tableau calculi model possible worlds and the accessibility relation between them by prefixes (either at the meta level [23] or at the object level [3]). Prefixed tableaux usually allow for straightforward completeness arguments but often require complicated blocking mechanisms [20, 16] for termination.

An alternative view of tableaux was developed by Pratt [26] and elaborated by Goré et al. [12, 14] for PDL. There, tableaux are seen as possibly cyclic graphs. Given a graph tableau, a corresponding (typically infinite) tree tableau can be obtained by a tree unfolding of the graph. A decision procedure based on graph tableaux conceptually consists of two stages. First, given an input formula s, the procedure constructs the tableau (graph) for s. Then, the procedure checks if the constructed graph contains an evident subtableau that contains s. While being presented as two successive steps in [26], the two stages are interleaved in [12, 14] for performance reasons. For temporal logics, the graph tableau approach is adopted by Manna and Wolper [22] and by Kesten et al. [19] (there also exist decision procedures for alternation-free μ-calculi based on graph tableaux [29] that are, however, not incremental in the sense of [19]). Graph tableaux are usually set up in a way that makes termination obvious, often allowing to obtain worst-case optimal decision procedures for expressive logics [26, 12, 14].

In this paper we propose a uniform treatment of a family of modal and hybrid logics by graph tableaux. To abstract away from propositional reasoning, we employ a clausal form developed for tree tableaux in our previous work [18] (note that our clausal form is different from the normal form by the same name used in [24, 13]). Our present investigations focus on graph tableaux as a way of establishing the small model property and decidability of modal logics, a role that has traditionally been filled by filtration. Filtration [4] was invented by Lemmon and Scott [21] and further developed by Segerberg [28] and, in a somewhat different form, by Gabbay [10]. Fischer and Ladner [8] were the first to apply filtration to a logic with eventualities. While being similarly elegant to filtration, graph tableaux offer an important advantage. Rather than just providing an upper bound on the size of a minimal model of a satisfiable formula, they provide a way of constructing such a model in a goal-directed way.

We demonstrate clausal graph tableaux on H_D^*, which is modal logic extended with nominals, eventualities and the difference operator. Nominals are formulas of the form x that hold exactly for the state x. Eventualities are formulas of the form $\Diamond^* s$ that hold for a state if it can reach in $n \geq 0$ steps a state satisfying the formula s. A difference formula $\mathsf{D}s$ holds for a state if there is a different state satisfying s. Nominals and the difference operator D equip modal logic with equality, a characteristic feature of hybrid logic [4, 2]. Eventualities extend modal logic with reflexive transitive closure and are an essential feature of PDL [8, 15] and temporal logics [25, 6, 7]. One can see H_D^* either as hybrid logic extended with eventualities or as stripped-down PDL extended with nominals and D. Due to the inductive nature of eventualities, H^*_D is not compact (consider $\Diamond^*\neg p$, p, $\Box p$, $\Box\Box p$, ...). The EXPTIME-hardness of H^*_D follows from Fischer and Ladner's proof for PDL [8] (see Blackburn et al. [4], Theorem 6.52). The method in this paper yields a NEXPTIME upper bound for H^*_D. This seems to be the first upper bound established for H^*_D.

In [18], we develop a clausal tree tableau calculus for H^* (modal logic with nominals and eventualities). While it is easy to give clausal tree tableaux for hybrid logic with D, we found it difficult to give an elegant clausal treatment of logics containing both eventualities and difference (such as H^*_D and $HPDL_D$) using tree tableaux. Moreover, the graph tableau approach has not been applied so far to logics with nominals. By adapting the clausal approach and the solutions developed for nominals in [18] to graph tableaux, we are able to give a satisfactory treatment of both eventualities and D within a single framework. Unlike the approach in [18], which works on a single branch of a tree tableau at a time, the present approach is fully deterministic, representing all possible choices within a single graph tableau. This allows us to share the computational costs necessary to deal with D across the tableau instead of paying them on every branch. Another advantage of graph tableaux over the approach in [18] is a significantly simpler soundness argument. The soundness of the calculus in [18] relies on an invariant of tableau branches, called straightness. To maintain the invariant, a technique reminiscent of blocking is used. The present approach does not rely on any such invariants for soundness. We believe that following the ideas in [17] the present approach scales to $HPDL_D$ (PDL extended with nominals and D).

We see the main contribution of the present paper in extending the graph tableau approach to hybrid logic with eventualities and difference, while at the same time giving the first goal-directed decision procedure for this logic. The use of a clausal form allows for an elegant presentation and simple, modular correctness arguments.

The paper is organized as follows. First, we introduce the approach on the basic modal logic with eventualities. Then we show how graph tableaux adapt to basic hybrid logic and hybrid logic with difference. Finally, we combine the treatment of eventualities with that of nominals and the difference operator.

2 Hybrid Logic with Eventualities and Difference

We define the syntax and semantics of the basic hybrid logic with eventualities and difference. We assume that two kind of names, called **nominals** and **predicates**, are given. Nominals (written x, y) denote states and predicates (written

p, q) denote sets of states. **Formulas** are defined as follows:

$$s ::= x \mid p \mid \neg s \mid s \wedge s \mid \Diamond s \mid \Diamond^* s \mid \mathsf{D}s$$

For simplicity we employ only a single transition relation. The extension of the approach to multimodal logic is straightforward. Formulas prefixed with the **diamond operators** $\Diamond$ and $\Diamond^*$ are called **diamond formulas** and formulas prefixed with the **difference operator** D are called **difference formulas**.

An **interpretation** $\mathcal{I}$ consists of the following components:

- A nonempty set $|\mathcal{I}|$ of **states**.
- A **transition relation** $\rightarrow_{\mathcal{I}} \subseteq |\mathcal{I}| \times |\mathcal{I}|$.
- A state $\mathcal{I}x \in |\mathcal{I}|$ for every nominal x.
- A set $\mathcal{I}p \subseteq |\mathcal{I}|$ for every predicate p.

The **satisfaction relation** $\mathcal{I}, X \vDash s$ between interpretations $\mathcal{I}$, states $X \in |\mathcal{I}|$, and formulas s is defined by induction on s:

$$\begin{aligned} \mathcal{I}, X \vDash x &\iff X = \mathcal{I}x & \mathcal{I}, X \vDash s \wedge t &\iff \mathcal{I}, X \vDash s \text{ and } \mathcal{I}, X \vDash t \\ \mathcal{I}, X \vDash p &\iff X \in \mathcal{I}p & \mathcal{I}, X \vDash \Diamond s &\iff \exists Y\colon X \rightarrow_{\mathcal{I}} Y \text{ and } \mathcal{I}, Y \vDash s \\ \mathcal{I}, X \vDash \neg s &\iff \text{not } \mathcal{I}, X \vDash s & \mathcal{I}, X \vDash \Diamond^* s &\iff \exists Y\colon X \rightarrow_{\mathcal{I}}^* Y \text{ and } \mathcal{I}, Y \vDash s \\ & & \mathcal{I}, X \vDash \mathsf{D}s &\iff \exists Y\colon X \neq Y \text{ and } \mathcal{I}, Y \vDash s \end{aligned}$$

$\rightarrow_{\mathcal{I}}^*$ denotes the reflexive transitive closure of $\rightarrow_{\mathcal{I}}$

Given a set A of formulas, we write $\mathcal{I}, X \vDash A$ if $\mathcal{I}, X \vDash s$ for all formulas $s \in A$. An interpretation $\mathcal{I}$ **satisfies** (or is a **model** of) a formula s or a set A of formulas if there is a state $X \in |\mathcal{I}|$ such that $\mathcal{I}, X \vDash s$ or, respectively, $\mathcal{I}, X \vDash A$. A formula s (a set A) is **satisfiable** if s (A) has a model.

The **complement** $\sim$ of a formula s is t if $s = \neg t$ and $\neg s$ otherwise. Note that $\sim\sim s = s$ if s is not a double negation. We use the notations $s \vee t := \neg(\sim s \wedge \sim t)$, $\Box s := \neg\Diamond\sim s$, $\Box^* s := \neg\Diamond^*\sim s$, and $\bar{\mathsf{D}}s := \neg\mathsf{D}\sim s$. Note that $\sim\Diamond p = \Box\neg p$ and $\sim\Diamond\neg p = \Box p$. Moreover, we define $\Diamond^+ s := \Diamond\Diamond^* s$ and $\Box^+ s := \Box\Box^* s$ (note that $\Box^+ s = \Box\Box^* s = \neg\Diamond\sim\neg\Diamond^*\sim s = \neg\Diamond\Diamond^*\sim s = \neg\Diamond^+\sim s$). An **eventuality** is a formula of the form $\Diamond^* s$ or $\Diamond^+ s$. All other diamond formulas are called **simple**.

We write H_{D}^* for the full logic and define several sublogics:

K	$p \mid \neg s \mid s \wedge s \mid \Diamond s$
K*	K extended with $\Diamond^* s$
H	K extended with x
H_{D}	H extended with $\mathsf{D}s$
H*	H extended with $\Diamond^* s$ (or K* extended with x)
H_{D}^*	H_{D} extended with $\Diamond^* s$ (or H* extended with $\mathsf{D}s$)

3 Clausal Form

We define a clausal form for our logic. The clausal form allows us to abstract from propositional reasoning and to focus on modal reasoning. Rather than committing to a particular clausal form, we make explicit the abstract properties of the clausal form that we need for our results. This makes our results more widely applicable since the abstract properties are general enough to allow for different clausal forms, in particular for clausal forms compatible with propositional optimizations like semantic branching [30]. A naive computable clausal form is then suggested in the proof of Proposition 3.3.

A **basic formula** is a formula of the form x, p, $\Diamond s$, or $\mathsf{D}s$. A **literal** is a basic formula or the complement of a basic formula. A **clause** (denoted by C, D) is a finite set of literals that contains no complementary pair. Note that since clauses contain no complementary literals, for every finite set A of basic formulas there are $3^{|A|}$ clauses C such that $C \subseteq A \cup \{\, \neg s \mid s \in A \,\}$. Clauses are interpreted conjunctively. **Satisfaction of clauses** (i.e., $\mathcal{I}, X \models C$) is a special case of satisfaction of sets of formulas (i.e., $\mathcal{I}, X \models A$), which was defined in §2. For instance, the clause $\{\Diamond p,\ \Box(\neg p \wedge q)\}$ is unsatisfiable. Note that every clause not containing literals of the form $\Diamond s$ or $\mathsf{D}s$ is satisfiable.

To deal with the difference operator, our approach assumes an injective function that assigns to every literal $\mathsf{D}s$ a nominal $x_{\mathsf{D}s}$. Intuitively, a nominal $x_{\mathsf{D}s}$ is supposed to denote a state that satisfies s, provided such a state exists. If it does, all states that are different from the one denoted by $x_{\mathsf{D}s}$ satisfy $\mathsf{D}s$. A **base** is a set A of basic formulas such that:

1. If $s \in A$ and t is a basic subformula of s, then $t \in A$.
2. If $s \in A$ and $\Diamond^* t$ is a subformula of s, then $\Diamond^+ t \in A$.
3. If $\mathsf{D}s \in A$, then $x_{\mathsf{D}s} \in A$.

While conditions (1) and (2) are required for all extensions of K*, (3) is only needed for the difference operator. A set A is called a **base of a formula** s if A is a base, contains every basic subformula of s and, additionally, $\Diamond^+ t \in A$ whenever $\Diamond^* t$ is a subformula of s. Note that in particular we have that every base of a formula $\Box^* s$ (i.e., $\neg\Diamond^* {\sim} s$) is also a base of $\Box^+ s$ (i.e., $\neg\Diamond^+ {\sim} s$). Intuitively, a base of a formula s contains all basic formulas that need to be evaluated (not necessarily all at the same state) to determine the truth value of s. A set A is a **base of a set of formulas** B if A is a base of every $s \in B$. Let A be a set of formulas. We write $\mathcal{B}A$ for the least base of A. It can be shown that $\mathcal{B}A$ is finite if A is finite, and that the size of $\mathcal{B}A$ is linear in the size of A, i.e., the sum of the sizes of the formulas occurring as elements of A (except for the nominals $x_{\mathsf{D}s}$, $\mathcal{B}s$ is a subset of the Fischer-Ladner closure of s; cf. [8, 15]).

The **support relation** $C \rhd s$ between clauses C and formulas s is defined by induction on s:

$$\begin{array}{rcl@{\qquad}rcl} C \rhd s & \iff & s \in C \quad \text{if } s \text{ is a literal} & C \rhd \neg\neg s & \iff & C \rhd s \\ C \rhd s \wedge t & \iff & C \rhd s \text{ and } C \rhd t & C \rhd s \vee t & \iff & C \rhd s \text{ or } C \rhd t \\ C \rhd \Diamond^* s & \iff & C \rhd s \text{ or } C \rhd \Diamond^+ s & C \rhd \Box^* s & \iff & C \rhd s \text{ and } C \rhd \Box^+ s \end{array}$$

We say C **supports** s if $C \rhd s$. We write $C \rhd A$ and say C **supports** A if $C \rhd s$ for every $s \in A$. Note that $C \rhd D \iff D \subseteq C$ if C and D are clauses.

Proposition 3.1 If $C \rhd A$ and $C \subseteq D$ and $B \subseteq A$, then $D \rhd B$.

Proposition 3.2 If $\mathcal{I}, X \models C$ and $C \rhd A$, then $\mathcal{I}, X \models A$.

A **DNF** (disjunctive normal form) is a function $\mathcal{D}$ that maps every finite set A of formulas to a finite set of clauses such that:

1. $\mathcal{I}, X \models A \iff \exists D \in \mathcal{D}A\colon\ \mathcal{I}, X \models D$.
2. $C \rhd A \iff \exists D \in \mathcal{D}A\colon\ D \subseteq C$.
3. If $C \in \mathcal{D}A$, then $C \subseteq \mathcal{B}A \cup \{\neg s \mid s \in \mathcal{B}A\}$.

Note that the third property of DNFs may equivalently be stated as follows: If $C \in \mathcal{D}A$, then $\mathcal{B}C \subseteq \mathcal{B}A$. Note further that, restricted to propositional logic, our notion of a DNF reduces to the common notion of a propositional DNF. Consider, for instance, a function $\mathcal{D}$ such that $\mathcal{D}\{p_1 \wedge (\neg p_2 \vee p_3)\} = \{\{p_1, \neg p_2\}, \{p_1, p_3\}\}$. Clearly, $\mathcal{D}$ is a DNF according to the above definition. By interpreting $\{\{p_1, \neg p_2\}, \{p_1, p_3\}\}$ as $(p_1 \wedge \neg p_2) \vee (p_1 \wedge p_3)$ we see that $\mathcal{D}$ indeed computes a DNF of $p_1 \wedge (\neg p_2 \vee p_3)$.

Proposition 3.3 There is a computable DNF.

Proof The definition of the support relation can be seen as a tableau-style decomposition procedure for formulas. The clauses of a DNF can be obtained with the literals the decomposition produces. The direction "$\Leftarrow$" of property (1) of DNFs follows with Proposition 3.2. Properties (2) and (3) of DNFs easily follow, respectively, from the definitions of the support relation and the base of a formula. ∎

For the rest of the paper, we fix some computable DNF $\mathcal{D}$. In our examples, in particular in Examples 4.3 and 5.3, we assume that $\mathcal{D}$ is defined as suggested in the proof of Proposition 3.3. In particular, we assume that $\mathcal{D}C = \{C\}$ for all clauses C, and $\mathcal{D}\{s \wedge t\} = \{\{s, t\}\}$ for every two non-complementary literals s, t.

The **request of a clause** C is $\mathcal{R}C := \{t \mid \Box t \in C\}$.

Proposition 3.4 If $\mathcal{I}, X \models C$ and $X \to_{\mathcal{I}} Y$, then $\mathcal{I}, Y \models \mathcal{R}C$.

4 Tableaux for K^*

To demonstrate the basic ideas of the graph tableau approach, we begin with a tableau system for K^*, the basic modal logic with eventualities. Hence, for the rest of the section, we restrict formulas to be of the form:

$$s ::= p \mid \neg s \mid s \wedge s \mid \Diamond s \mid \Diamond^* s$$

Basic formulas, literals and clauses are restricted accordingly.

A **claim** is a pair $C^{\Diamond s}$ such that $\Diamond s \in C$. Given a formula s and a set A, we write $A\,;s$ for the set $A \cup \{s\}$. A **link** is a triple $C^{\Diamond s}D$ such that $C^{\Diamond s}$ is a claim and either

1. $\Diamond s$ is not an eventuality and $D \in \mathcal{D}(\mathcal{R}C\,;s)$, or
2. $s = \Diamond^* t$ and $D \in \mathcal{D}(\mathcal{R}C\,;t) \cup \mathcal{D}(\mathcal{R}C\,;\Diamond s)$.

A **tableau** is a finite non-empty set T of clauses and links such that $C, D \in T$ whenever $C^s D \in T$. A tableau T is **complete** if $C^{\Diamond s}D \in T$ whenever $\Diamond s \in C \in T$ and $C^{\Diamond s}D$ is a link. Given a tableau T, we define $\mathcal{B}T$ as the least set A that is a base of all $C \in T$. It is easily seen that $\mathcal{B}T = \bigcup\{\, \mathcal{B}C \mid C \in T\,\}$.

The architecture of a decision procedure based on graph tableaux is as follows. Given a clause C, we construct a tableau T that contains C and satisfies certain completeness criteria. The size of T will be exponentially bounded in the size of the base of C. If C is satisfiable, T will contain a subtableau that syntactically describes a model of C. We call such subtableaux evident. The existence of evident subtableaux is decidable since T is finite. Given the decision procedure for clauses, a procedure for formulas is immediate by property (1) of DNFs.

Proposition 4.1 If $C^s D$ is a link, then $\mathcal{B}D \subseteq \mathcal{B}C$.

For every clause C, we can construct a complete tableau T containing C by closing the initial tableau $\{C\}$ under the following **completion rule**:

$$\frac{D}{E,\ D^{\Diamond s}E}\ \Diamond s \in D,\ D^{\Diamond s}E \text{ link}$$

The closure is finite since for all clauses D added by the construction we have $\mathcal{B}D \subseteq \mathcal{B}C$ (follows by Proposition 4.1). Hence, the number of clauses in T is bounded by $3^{|\mathcal{B}C|}$. The number of links in T is then bounded by $|\mathcal{B}C| \cdot 9^{|\mathcal{B}C|}$. Clearly, $C \in T$ and $\mathcal{B}T = \mathcal{B}C$.

Proposition 4.2 For every clause C there is a complete tableau T such that $C \in T$, $\mathcal{B}T = \mathcal{B}C$, and $|T| = 3^{|\mathcal{B}C|} + |\mathcal{B}C| \cdot 9^{|\mathcal{B}C|}$.

A **run for $C^{\Diamond^+ s}$ in T** is a sequence of clauses $C_1 \ldots C_n$ such that:

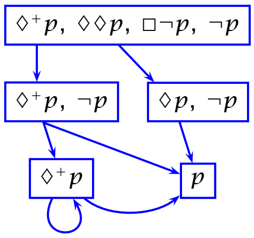
1. $C_1 = C$.
2. $\forall i \in [1, n-1]\colon\ C_i{}^{\Diamond^+ s} C_{i+1} \in T$.
3. $C_n \rhd s$.

We call a tableau evident if every claim has an outgoing link and every claim of the form $C^{\Diamond^+ s}$ has a run. More precisely: A tableau T is **evident** if:

1. $\forall \Diamond s \in C \in T\ \exists D\colon\ C^{\Diamond s} D \in T$.
2. T has a run for $C^{\Diamond^+ s}$ whenever $\Diamond^+ s \in C \in T$.

An interpretation **satisfies** (or is a **model** of) a tableau if it satisfies all of its clauses.

Example 4.3 Consider the complete tableau T for the clause $C_0 = \{\Diamond^+ p,\ \Diamond\Diamond p,\ \Box\neg p,\ \neg p\}$ obtained with the above completion rule.



Links of the form $C^s D$ are represented graphically by arrows going from the formula s in the clause C to the clause D. Consider the claim $C_0{}^{\Diamond^+ p}$. Since $\Diamond^+ p$ is an eventuality and $\mathcal{R}C_0 = \{\neg p\}$, a triple $C_0{}^{\Diamond^+ p} D$ is a link if and only if $D \in \mathcal{D}\{p, \neg p\} \cup \mathcal{D}\{\Diamond^+ p, \neg p\} = \emptyset \cup \{\{\Diamond^+ p, \neg p\}\} = \{\{\Diamond^+ p, \neg p\}\}$. Hence, $\{\Diamond^+ p, \neg p\}$ is the only clause and $C_0{}^{\Diamond^+ p}\{\Diamond^+ p, \neg p\}$ the only link that is added by the completion rule for the claim $C_0{}^{\Diamond^+ p}$. Note that T is evident since every claim has at least one outgoing link and the clause $\{p\}$ is reachable from every diamond $\Diamond^+ p$. In particular, $\{\Diamond^+ p, \neg p\}\{\Diamond^+ p\}\{p\}$ is a run for $\{\Diamond^+ p, \neg p\}^{\Diamond^+ p}$ in T. $\Box$

Theorem 4.4 (Model Existence) Evident tableaux have finite models.

Proof Let T be an evident tableau. We choose an interpretation $\mathcal{I}$ such that:

- $|\mathcal{I}| = \{\, C \mid C \in T \,\}$
- $C \rightarrow_{\mathcal{I}} D \iff \exists s\colon\ C^{\Diamond s} D \in T$
- $C \in \mathcal{I}p \iff p \in C$

Clearly, $|\mathcal{I}|$ is finite since T is finite.

We show $\forall s\ \forall C \in T\colon\ C \rhd s \implies \mathcal{I}, C \models s$ by induction on s. Let $C \in T$ and $C \rhd s$. We show $\mathcal{I}, C \models s$ by case analysis. The argument is straightforward except possibly for the cases $s = \Diamond^* t$ and $s = \Box^* t$.

Let $s = \Diamond^* t$. Since $C \triangleright s$, we have either $C \triangleright t$ or $C \triangleright \Diamond^+ t$. If $C \triangleright t$, then $\mathcal{I}, C \models t$ by the inductive hypothesis, and the claim follows. Otherwise, let $C \triangleright \Diamond^+ t$. Then $\Diamond^+ t \in C \in T$. By the second evidence condition we know that there is a run for $C^{\Diamond^+ t}$ in T. Thus $C \to_{\mathcal{I}}^* D$ and $D \triangleright t$ for some clause $D \in \Gamma$. Hence $\mathcal{I}, D \models t$ by the inductive hypothesis. The claim follows.

Let $s = \Box^* t$. Let $C = C_1 \to_{\mathcal{I}} \ldots \to_{\mathcal{I}} C_n$. We show $\mathcal{I}, C_n \models t$ by induction on n. If $n = 1$, we have $C_n \models s$ by assumption. Hence $C_n \models t$ and the claim follows by the outer inductive hypothesis. If $n > 1$, we have $s \in \mathcal{R}C_1$ since $\Box s \in C_1$ since $C_1 \triangleright \Box s$ since $C_1 \triangleright s$. Thus $C_2 \triangleright s$ and the claim follows by the inner inductive hypothesis. ■

A tableau T' is a **subtableau** of a tableau T if $T' \subseteq T$.

Theorem 4.5 (Evidence) Let T be a complete tableau and $C \in T$. If C is satisfiable, then there is an evident subtableau of T containing C.

Proof (Sketch) Let T and C be as required, and let $\mathcal{I}$ be a model of C. We define U to consist of the clauses of T that are satisfied by $\mathcal{I}$, and the edges $D^s E \in T$ such that $\{D, E\} \subseteq U$. It is easily seen that U is a subtableau of T and that $C \in U$, so it remains to show that U is evident. Showing evidence condition (1) is straightforward. As for condition (2), observe that whenever we have $\mathcal{I}, X_0 \models \Diamond^+ s$, this means that there is a sequence of states $X_1 \ldots X_n$ ($n \geq 1$) such that, for all $i \in [1, n]$, $X_{i-1} \to_{\mathcal{I}} X_i$, and $\mathcal{I}, X_n \models s$. Now, to show condition (2), we show that, given a claim ${C_0}^{\Diamond^+ s}$ and assuming $\mathcal{I}, X_0 \models C_0$, the sequence $X_1 \ldots X_n$ can be "projected" onto clauses $C_1 \ldots C_n$ of U such that, for all $i \in [1, n]$, we have (a) $\mathcal{I}, X_i \models C_i$, (b) $X_{i-1} \to_{\mathcal{I}} X_i$ implies ${C_{i-1}}^{\Diamond^+ s} C_i \in U$, and (c) $C_n \triangleright s$. Clearly, this implies that $C_0 C_1 \ldots C_n$ is a run for ${C_0}^{\Diamond^+ s}$, which suffices for the claim. See Appendix for details. ■

5 Tableaux for $\mathsf{H_D}$

Before we proceed to the full logic, let us develop graph tableaux for $\mathsf{H_D}$. This will allow us to introduce the machinery needed for nominals without the complications that are added by eventualities. To account for nominals, we will allow links to clauses that are larger than those given by the DNF of the diamond formula and the request of the source clause. To reduce redundancy in complete graph tableaux, we will introduce the notion of a link closure and require a complete tableau to contain an evident subtableau in its link closure. We will point out which parts of the construction are needed for hybrid logic in general and which are there specifically to account for the difference operator. The formulas of $\mathsf{H_D}$ look as follows:

$$s ::= x \mid p \mid \neg s \mid s \wedge s \mid \Diamond s \mid \mathsf{D}s$$

To deal with nominals, we need to adapt the definition of links. A **link** is a triple $C^{\Diamond s}D$ such that $C^{\Diamond s}$ is a claim and $D \rhd \mathcal{R}C\,;s$. A link $C^{\Diamond s}D$ is called **minimal** if $D \in \mathcal{D}(\mathcal{R}C\,;s)$ (i.e., minimal links are precisely the special kind of links used in § 4). Proposition 4.1 adapts as follows:

Proposition 5.1 If C^sD is a minimal link, then $\mathcal{B}D \subseteq \mathcal{B}C$.

Given the new definition of links, tableaux are defined as before. A tableau T is **complete** if:

1. If $\Diamond s \in C \in T$ and $C^{\Diamond s}D$ is a minimal link, then $C^{\Diamond s}D \in T$.
2. If $C \in T$ and $x \in \mathcal{B}C$, then $\{x\} \in T$.
3. If $C, D \in T$, $x \in C \cap D$, and $C \cup D$ is a clause, then $C \cup D \in T$.
4. If $\mathsf{D}s \in C \in T$, $x_{\mathsf{D}s} \notin C$, and $D \in \mathcal{D}\{x_{\mathsf{D}s},\ s\}$, then $D \in T$.
5. If $\mathsf{D}s \in C \in T$, $x_{\mathsf{D}s} \in C$, and $D \in \mathcal{D}\{\neg x_{\mathsf{D}s},\ s\}$, then $D \in T$.
6. If $\bar{\mathsf{D}}s \in C \in T$, $D \in T$, $D \not\rhd s$, and $C \cup D$ is a clause, then $C \cup D \in T$.
7. If $\bar{\mathsf{D}}s \in C \in T$, $D \in T$, $D \not\rhd s$, and $E \in \mathcal{D}(D\,;s)$, then $E \in T$.

Note that to obtain a complete system for H, only the first three of the completeness criteria are needed. The last four criteria are there exclusively to deal with D and its dual. Recall that the idea behind the completion rules is to generate enough clauses so that we can select an evident subset. Criterion (1) is the obvious adaptation of the completeness criterion from § 4. Criteria (2) and (3) are motivated by the semantics of nominals. Every nominal x has to denote some state that, obviously, satisfies $\{x\}$. Moreover, if two clauses are satisfied by the same model and have a nominal in common, then they hold in the same state of the model, and hence their union is also satisfiable. For (4) and (5), recall that a nominal $x_{\mathsf{D}s}$ is assumed to denote a state that satisfies s (provided such a state exists). So, if $\mathsf{D}s$ is satisfiable anywhere in a model, then $\{x_{\mathsf{D}s},\ s\}$ is satisfiable, and if $\{\mathsf{D}s,\ x_{\mathsf{D}s}\}$ is satisfiable, then so is $\{\neg x_{\mathsf{D}s},\ s\}$. Criteria (6) and (7) are motivated as follows. If $\bar{\mathsf{D}}s$ holds in a state X, then every state that is distinct from X satisfies s. So, every given state Y must either be equal to X (cf. (6)) or satisfy s (cf. (7)).

Note that, analogously to the above tableau construction rule in § 4, the above completeness criteria can be interpreted as tableau rules (called **completion rules**). So, for instance, criterion (1) and (3) translate to, respectively:

1) $\dfrac{C}{D,\ C^{\Diamond s}D}\ \ \Diamond s \in C,\ C^{\Diamond s}D$ minimal link 3) $\dfrac{C,\ D}{C \cup D}\ \ C \cup D$ clause, $\exists x\colon x \in C \cap D$

Analogously to the argument in § 4, we obtain:

Proposition 5.2 For every clause C there is a complete tableau T such that $C \in T$, $\mathcal{B}T = \mathcal{B}C$, and $|T| = 3^{|\mathcal{B}C|} + |\mathcal{B}C| \cdot 9^{|\mathcal{B}C|}$.

We define $A^T := A \cup \{\, s \mid \exists x \in A\ \exists C \in T\colon\ x \in C \text{ and } s \in C \,\}$. A tableau T is **evident** if:

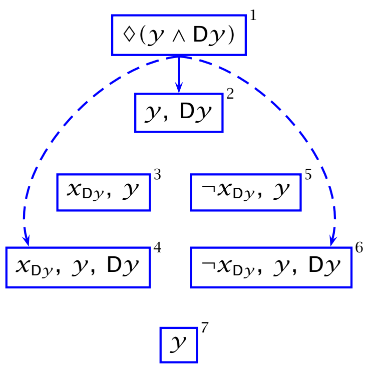
1. $\forall \Diamond s \in C \in T\ \exists D\colon\ C^{\Diamond s}D \in T$.
2. If $C \in T$ and $x \in \mathcal{B}C$, then there is some $D \in T$ such that $x \in D$.
3. If $C \in T$, then $C = C^T$.
4. If $\mathsf{D}s \in C \in T$, then there is some $D \in T$ such that $D \neq C$ and $D \rhd s$.
5. If $\bar{\mathsf{D}}s \in C \in T$, then, for all $D \in T$ such that $D \neq C$, we have $D \rhd s$.

Again, for hybrid logic without D we only need the first three conditions.

In the presence of nominals and D, clauses introduced by the rule for criterion (1) may be too small to be included in the evident tableau (because of evidence condition (3) or (5)). While criteria (3), (6) and (7) will ensure that the complete tableau contains the larger clauses that are needed, they will add no links to the new clauses. Instead, we add the required links in a uniform way by defining the **link closure** $\hat{T}$ of a tableau T as

$$\hat{T} := T \cup \{\, C^s D \mid \exists E\colon\ E \subseteq D \text{ and } C^s E \in T \,\}$$

Example 5.3 Consider the following complete tableau T for the unsatisfiable clause $\{\Diamond(y \wedge \mathsf{D}y)\}$.



The numbers indicate in which order the clauses are introduced by the completion rules. So, clause (2) is derived from (1) by the rule corresponding to completeness criterion (1), clause (3) follows from (2) by the rule for criterion (4), clause (4) follows from (2) and (3) by the rule for criterion (3), clause (5) follows from (4) by the rule for criterion (5), clause (6) follows from (2) and (5) by the rule for criterion (3), and clause (7) follows by the rule for criterion (2) applied to any of the preceding clauses. Note that the rule for criterion (3) does not apply to clauses (4) and (5) since their union is not a clause. The dashed arrows stand for the additional links in the link closure $\hat{T}$.

The tableau $\hat{T}$ contains no evident subtableau that contains $\{\Diamond(y \wedge \mathsf{D}y)\}$ (i.e., clause (1)). By evidence condition (1), an evident subtableau of $\hat{T}$ containing clause (1) would also have to contain either (2), (4) or (6), all of which contain the nominal y and the formula $\mathsf{D}y$. Then, by evidence condition (4), the subtableau would have to contain a second clause containing y. However, having two distinct clauses that contain the same nominal contradicts evidence condition (3). □

As before for K*, one of our goals will be showing that complete tableaux for satisfiable clauses have evident subtableaux (now modulo link closure). This also explains why we introduce nominals $x_{\mathsf{D}s}$. We need them to ensure that subtableaux of a complete tableau satisfy evidence condition (4). Assume we simplified completeness criteria (4) and (5) to:

$$\text{If } \mathsf{D}s \in C \in T \text{ and } D \in \mathcal{D}\{s\}, \text{ then } D \in T.$$

Then $T := \{\{\bar{\mathsf{D}}\mathsf{D}p,\ \mathsf{D}p,\ \neg p\},\ \{\mathsf{D}p,\ p\},\ \{p\}\}$ would be a complete tableau. Moreover, $\hat{T} = T$. Although all clauses of T are satisfiable, T contains no evident subtableau containing $\{\bar{\mathsf{D}}\mathsf{D}p,\ \mathsf{D}p,\ \neg p\}$.

Theorem 5.4 (Model Existence) Evident tableaux have finite models.

Proof Let T be an evident tableau. By evidence conditions (2) and (3), for every $x \in \mathcal{B}T$ we have a unique clause $C \in T$ such that $x \in C$. We choose an interpretation $\mathcal{I}$ as in the proof of Theorem 4.4 such that additionally, for all $x \in \mathcal{B}T$, $\mathcal{I}x = C \iff x \in C$. Again, $|\mathcal{I}|$ is finite as so is T. We show $\forall s\ \forall C \in T\colon\ C \rhd s \implies \mathcal{I}, C \models s$ by induction on s. The verification of the individual cases is straightforward. ■

Theorem 5.5 (Evidence) Let T be a complete tableau and let $C \in T$ be such that, for all $t \in C$ and $\mathsf{D}s \in \mathcal{B}T$, $x_{\mathsf{D}s}$ does not occur in t. If C is satisfiable, then there is an evident subtableau U of $\hat{T}$ and a clause $D \in U$ such that $C \subseteq D$.

Proof (Sketch) Let T and C be as required, and let $\mathcal{I}$ be a model of C (with some additional constraints). We define U to consist of the maximal clauses among all clauses of T that are satisfied by $\mathcal{I}$. As the edges of U we take all $D^s E \in \hat{T}$ such that $\{D,\ E\} \subseteq U$. One can show that U has the desired properties. See Appendix for details. ■

6 Tableaux for H^* and H^*_D

Now that we know how graph tableaux look for eventualities and nominals in isolation, let us approach their combinations, H^* and H^*_D. The addition of D to H turns out to be particularly straightforward since the cases for D in the proofs of evidence and model existence for H_D can be treated orthogonally from the rest of the respective arguments. For H^*, this is no longer the case. While model existence is still straightforward, given a meaningful definition of an evident tableau, D significantly complicates the evidence proof when combined with eventualities.

Links and minimal links are now defined as follows. A **link** is a triple $C^{\Diamond s}D$ such that $C^{\Diamond s}$ is a claim and either

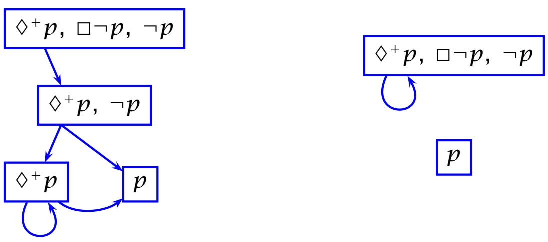
1. $\Diamond s$ is not an eventuality and $D \triangleright \mathcal{R}C\,;s$, or
2. $s = \Diamond^* t$ and $D \triangleright \mathcal{R}C\,;t$ or $D \triangleright \mathcal{R}C\,;\Diamond s$.

A link $C^{\Diamond s}D$ is called **minimal** if

1. $\Diamond s$ is not an eventuality and $D \in \mathcal{D}(\mathcal{R}C\,;s)$, or
2. $s = \Diamond^* t$ and $D \in \mathcal{D}(\mathcal{R}C\,;t) \cup \mathcal{D}(\mathcal{R}C\,;\Diamond s)$.

Given the new definitions, tableaux are defined as in § 4. The completeness criteria, completion rules and the link closure for H^*_D (resp., H^*) look exactly the same as for H_D (resp., H). Also, Propositions 5.1 and 5.2 are easy to re-prove for the new definitions.

As it turns out, taking maximal clauses to obtain evidence in the presence of nominals, as it is done in the proof of Theorem 5.5, may destroy runs that are necessary for evidence in the presence of eventualities. Consider the following tableau (a):



(a) (b)

The tableau (a) is complete and satisfies the evidence conditions for K^*. All clauses of (a) are satisfiable. However, the maximal clauses construction in the proof of Theorem 5.5 produces the tableau (b), which does not satisfy evidence condition (2) for K^*. In the absence of D we can solve the problem by adapting the construction from [18] to graph tableaux. To cope with D, however, the approach needs considerable refinement.

The evidence conditions for H^*_D are obtained by taking the union of the conditions for K^* and H_D. A tableau T is **evident** if:

1. $\forall \Diamond s \in C \in T\ \exists D\colon\ C^{\Diamond s}D \in T$.
2. T has a run for $C^{\Diamond^+ s}$ whenever $\Diamond^+ s \in C \in T$.
3. If $C \in T$ and $x \in \mathcal{B}C$, then there is some $D \in T$ such that $x \in D$.
4. If $C \in T$, then $C = C^T$.
5. If $\mathsf{D}s \in C \in T$, then there is some $D \in T$ such that $D \neq C$ and $D \rhd s$.
6. If $\bar{\mathsf{D}}s \in C \in T$, then, for all $D \in T$ such that $D \neq C$, we have $D \rhd s$.

Theorem 6.1 (Model Existence) Evident tableaux have finite models.

Proof Let T be an evident tableau. We choose $\mathcal{I}$ as in the proof of Theorem 5.4 and show $\forall s\ \forall C \in T\colon\ C \rhd s \implies \mathcal{I}, C \models s$ by induction on s. The case distinction on the shape of s proceeds as in the proofs of Theorems 4.4 and 5.4. All cases but $s = \Diamond^* t$ and $s = \Box^* t$ proceed exactly as in the proof of Theorem 5.4. The cases $s = \Diamond^* t$ and $s = \Box^* t$ that are not covered by the proof of Theorem 5.4 proceed as in the proof of Theorem 4.4. ■

As for evidence, let us begin with with an outline of the proof for H^*.

Theorem 6.2 (Evidence for H*) Let T be a complete tableau for H^* and let $C \in T$. If C is satisfiable, then there is an evident subtableau U of $\hat{T}$ and a clause $D \in U$ such that $C \subseteq D$.

Proof Let T and C be as required, and let $\mathcal{I}$ be a model of C. Let $T' := \{\, E \in T \mid \mathcal{I} \text{ satisfies } E \,\}$. We define U such that:

1. $D \in U \ :\Longleftrightarrow\ D \in T'$ and $D = D^{T'}$.
2. $D^{\Diamond s}E \in U \ :\Longleftrightarrow\ D^{\Diamond s}E \in \hat{T}$ and $\{D,\ E\} \subseteq U$.

It is straightforward to verify that U is an evident subtableau of $\hat{T}$ that contains a superclause of C. ■

Unfortunately, the construction in the proof of Theorem 6.2 for selecting an evident subtableau does not work in the presence of D. The problem is caused by the evidence condition for formulas $\bar{\mathsf{D}}s$ (condition (6)). Consider the complete tableau $T := \{\{\bar{\mathsf{D}}p\},\ \{p\},\ \{\bar{\mathsf{D}}p,\ p\}\}$. Let $\mathcal{I}$ be an interpretation such that $|\mathcal{I}| = \mathcal{I}p = \{X\}$. Clearly, $\mathcal{I}$ satisfies all clauses of T. Since T contains no nominals, the construction in the proof of Theorem 6.2 for $\mathcal{I}$ yields $U = T$. However, T does not satisfy evidence condition (6) since $\bar{\mathsf{D}}p \in \{\bar{\mathsf{D}}p,\ p\} \in T$ but also $\{\bar{\mathsf{D}}p\} \in T$ (clearly, $\{\bar{\mathsf{D}}p\} \not\rhd p$). Note that although U is not evident, it still contains evident subtableaux ($\{\{\bar{\mathsf{D}}p\}\}$, $\{\{\bar{\mathsf{D}}p,\ p\}\}$ and $\{\{p\},\ \{\bar{\mathsf{D}}p,\ p\}\}$). As we noted in the beginning of the section, it is now not possible to take the maximal clauses of

U since this will in general destroy the evidence of eventualities (condition (2)). In the following, we will demonstrate how we can select an evident subtableau of U while preserving condition (2).

A key observation is that for every formula $\bar{\mathsf{D}}s$ that holds in some state $X \in |\mathcal{I}|$, we either have that s holds everywhere in $\mathcal{I}$, or that X is the unique state satisfying $\bar{\mathsf{D}}s$ and all other states satisfy s. Hence, for every formula $\bar{\mathsf{D}}s$ for which evidence condition (6) is violated (in a tableau T satisfied by $\mathcal{I}$), we can establish (6) by removing all clauses that do not support s (if s holds everywhere) or all such clauses except one (otherwise). In the latter case, we can select the remaining clause to be the largest clause containing $\bar{\mathsf{D}}s$. Since $\mathcal{I}$ satisfies T and X is the unique state in $\mathcal{I}$ satisfying $\bar{\mathsf{D}}s$, none of the clauses supporting s will contain $\bar{\mathsf{D}}s$, which guarantees that (6) is satisfied in the resulting tableau.

Lemma 6.3 Let T be a complete tableau and $\mathcal{I}$ an interpretation. Let

$$T' := \{\, C \in T \mid \mathcal{I} \text{ satisfies } C \,\}$$

Let $\bar{\mathsf{D}}s_1 \ldots \bar{\mathsf{D}}s_n$ be an injective enumeration of the set $\{\, \bar{\mathsf{D}}s \mid \bar{\mathsf{D}}s \in C \in T' \,\}$. Let $T'_0 := \{\, C \in T' \mid C = C^{T'} \,\}$. For all $i \in [1, n]$ we construct a set T'_i from T'_{i-1} as follows:

- If $\forall X \in |\mathcal{I}|\colon\ \mathcal{I}, X \vDash s_i$, then $T'_i := \{\, C \in T'_{i-1} \mid C \rhd s_i \,\}$.
- Otherwise, $T'_i := \{\, C \in T'_{i-1} \mid C \rhd s_i \,\};\bigcup\{\, C \in T'_{i-1} \mid \bar{\mathsf{D}}s_i \in C \,\}$.

Then, for all $i \in [1, n]$:

1. If $C \in T'_{i-1} \cap T'_i$, $D \in T'_{i-1}$, and $C \subseteq D$, then $D \in T'_i$.
2. If $C \in T'_{i-1}$ and $\mathcal{I}, X \vDash C$, then $C \subseteq D$ for some $D \in T'_i$ such that $\mathcal{I}, X \vDash D$.
3. $T'_i \subseteq T'_{i-1}$.
4. If $C \in T'_i$, then $C = C^{T'_i}$ (i.e., T'_i satisfies evidence condition (4)).
5. Let $j \in [1, i]$, $\bar{\mathsf{D}}s_j \in C \in T'_i$, and $D \in T'_i$ such that $D \neq C$. Then $D \rhd s_j$ (i.e., T'_i satisfies evidence condition (6) restricted to $\bar{\mathsf{D}}s_1, \ldots, \bar{\mathsf{D}}s_i$).

Proof See Appendix. ■

Theorem 6.4 (Evidence) Let T be a complete tableau and let $C \in T$ be such that, for all $t \in C$ and $\mathsf{D}s \in \mathcal{B}T$, $x_{\mathsf{D}s}$ does not occur in t. If C is satisfiable, then there is an evident subtableau U of $\hat{T}$ and a clause $D \in U$ such that $C \subseteq D$.

Proof (Sketch) Let T and C be as required, and let $\mathcal{I}$ be a model of C (with some additional constraints). Let T', $\mathsf{D}s_1 \ldots \mathsf{D}s_n$, $T'_1, \ldots, T'_n$ be defined from T and $\mathcal{I}$ as in Lemma 6.3. We define:

$$U := T'_n \cup \{\, D^s E \in \hat{T} \mid \{D,\ E\} \subseteq T'_n \,\}$$

Evidence conditions (4) and (6) hold by Lemma 6.3 (4,5). To show condition (2), we use the same basic technique as for K* (see the proof of Theorem 4.5). The argument is now more complex since not every clause from T that is satisfied by some state in $\mathcal{I}$ is still there in U. To show that we can still match every "witness sequence" $X_0 \ldots X_n$ by a run $C_0 \ldots C_n$, Lemma 6.3 (2) plays a crucial role. Lemma 6.3 (2) implies that for every satisfiable clause C in T there exists a superclause D in U that holds in the same state as C. Together with Lemma 6.3 (3), asserting that U is a subset of T with respect to clauses, Lemma 6.3 (2) is also central for showing evidence conditions (1), (3) and (5). See Appendix for details.
■

7 Conclusion

The paper presents the first goal-directed decision procedure for hybrid logic with eventualities and difference. A naive two-phase implementation of the procedure seems straightforward. Given an input clause C, we first compute a complete tableau containing C. This step takes at most deterministic exponential time in the size of the input (more precisely, in the size of the base of the input). To determine whether C is satisfiable, it then remains to check for the existence of an evident subtableau containing C. Naively, this can be done by repeatedly guessing candidate subtableaux and then checking their evidence. While the non-deterministic running time for the second phase is polynomial in the size of the complete tableau, because of the guessing, the deterministic algorithm is exponential. Hence, the combined procedure is in NEXPTIME, allowing for implementations with doubly exponential complexity. Based on results for related logics [1, 27], we conjecture H_D^* to be EXPTIME-complete. To reduce the complexity of our procedure to EXPTIME, provided this is possible at all, more work is needed. A promising direction is developing a polynomial algorithm for the second phase of the procedure, possibly following the ideas of [26]. The main complication here is that the procedure in [26] relies on the assumption that for every satisfiable formula s there exists a unique largest evident subtableau containing s. In our case, this assumption does not hold. In the presence of nominals, a complete tableau may contain several evident subtableaux whose union is not evident. Following [12, 14], one could also interleave the first and the second phase of the procedure so as to allow early pruning of unsatisfiable clauses. Compared to interleaved procedures for nominal-free logics [12, 14], such a procedure would have to deal with an additional difficulty, namely the link closure, which can contain considerably more links than the complete tableau.

References

[1] Carlos Areces, Patrick Blackburn, and Maarten Marx. The computational complexity of hybrid temporal logics. *L. J. IGPL*, 8(5):653–679, 2000.

[2] Carlos Areces and Balder ten Cate. Hybrid logics. In Blackburn et al. [5], pages 821–868.

[3] Patrick Blackburn. Internalizing labelled deduction. *J. Log. Comput.*, 10(1):137–168, 2000.

[4] Patrick Blackburn, Maarten de Rijke, and Yde Venema. *Modal Logic*. Cambridge University Press, 2001.

[5] Patrick Blackburn, Johan van Benthem, and Frank Wolter, editors. *Handbook of Modal Logic*, volume 3 of *Studies in Logic and Practical Reasoning*. Elsevier, 2007.

[6] E. Allen Emerson and Edmund M. Clarke. Using branching time temporal logic to synthesize synchronization skeletons. *Sci. Comput. Programming*, 2(3):241–266, 1982.

[7] E. Allen Emerson and Joseph Y. Halpern. "Sometimes" and "not never" revisited: On branching versus linear time temporal logic. *J. ACM*, 33(1):151–178, 1986.

[8] Michael J. Fischer and Richard E. Ladner. Propositional dynamic logic of regular programs. *J. Comput. System Sci.*, pages 194–211, 1979.

[9] Melvin Fitting. *Proof Methods for Modal and Intuitionistic Logics*. Reidel, 1983.

[10] Dov M. Gabbay. Selective filtration in modal logic, Part A. Semantic tableaux method. *Theoria*, 36(3):323–330, 1970.

[11] Jürgen Giesl and Reiner Hähnle, editors. *IJCAR 2010*, volume 6173 of *LNCS*. Springer, 2010.

[12] Rajeev Goré and Linh Anh Nguyen. EXPTIME tableaux with global caching for description logics with transitive roles, inverse roles and role hierarchies. In Nicola Olivetti, editor, *TABLEAUX 2007*, volume 4548 of *LNCS*, pages 133–148. Springer, 2007.

[13] Rajeev Goré and Linh Anh Nguyen. Clausal tableaux for multimodal logics of belief. *Fund. Inform.*, 94(1):21–40, 2009.

[14] Rajeev Goré and Florian Widmann. Optimal tableaux for propositional dynamic logic with converse. In Giesl and Hähnle [11], pages 225–239.

[15] David Harel, Dexter Kozen, and Jerzy Tiuryn. *Dynamic Logic*. The MIT Press, 2000.

[16] Ian Horrocks, Ullrich Hustadt, Ulrike Sattler, and Renate Schmidt. Computational modal logic. In Blackburn et al. [5], pages 181–245.

[17] Mark Kaminski and Gert Smolka. Clausal tableaux for hybrid PDL. Technical report, Saarland University, 2010.

[18] Mark Kaminski and Gert Smolka. Terminating tableaux for hybrid logic with eventualities. In Giesl and Hähnle [11], pages 240–254.

[19] Yonit Kesten, Zohar Manna, Hugh McGuire, and Amir Pnueli. A decision algorithm for full propositional temporal logic. In Costas Courcoubetis, editor, *CAV'93*, volume 697 of *LNCS*, pages 97–109. Springer, 1993.

[20] Saul A. Kripke. Semantical analysis of modal logic I: Normal modal propositional calculi. *Z. Math. Logik Grundlagen Math.*, 9:67–96, 1963.

[21] Edward J. Lemmon and Dana Scott. *The 'Lemmon Notes': An Introduction to Modal Logic*. Blackwell, 1977.

[22] Zohar Manna and Pierre Wolper. Synthesis of communicating processes from temporal logic specifications. *ACM TOPLAS*, 6(1):68–93, 1984.

[23] Fabio Massacci. Single step tableaux for modal logics. *J. Autom. Reasoning*, 24(3):319–364, 2000.

[24] Linh Anh Nguyen. A new space bound for the modal logics K4, KD4 and S4. In Mirosław Kutyłowski, Leszek Pacholski, and Tomasz Wierzbicki, editors, *MFCS'99*, volume 1672 of *LNCS*, pages 321–331, 1999.

[25] Amir Pnueli. The temporal logic of programs. In *Proc. 18th Annual Symp. on Foundations of Computer Science (FOCS'77)*, pages 46–57. IEEE Computer Society Press, 1977.

[26] Vaughan R. Pratt. A near-optimal method for reasoning about action. *J. Comput. System Sci.*, 20(2):231–254, 1980.

[27] Ulrike Sattler and Moshe Y. Vardi. The hybrid μ-calculus. In Rajeev Goré, Alexander Leitsch, and Tobias Nipkow, editors, *IJCAR 2001*, volume 2083 of *LNCS*, pages 76–91. Springer, 2001.

[28] Krister Segerberg. *An Essay in Classical Modal Logic*. Number 13 in Filosofiska Studier. University of Uppsala, 1971.

[29] Yoshinori Tanabe, Koichi Takahashi, and Masami Hagiya. A decision procedure for alternation-free modal μ-calculi. In Carlos Areces and Robert Goldblatt, editors, *Advances in Modal Logic*, volume 7, pages 341–362. College Publications, 2008.

[30] Dmitry Tsarkov, Ian Horrocks, and Peter F. Patel-Schneider. Optimizing terminological reasoning for expressive description logics. *J. Autom. Reasoning*, 39(3):277–316, 2007.

Appendix

Proposition 7.1 uses the notion of links as defined in § 4.

Proposition 7.1 Let $\Diamond s \in C$ and let $\mathcal{I}$ satisfy C. Then there is a link $C^{\Diamond s}D$ such that $\mathcal{I}$ satisfies D.

Proof Let $\Diamond s \in C$ and let $\mathcal{I}$ satisfy C. We distinguish two cases:

1. s is not an eventuality. Clearly, $\mathcal{I}$ satisfies $\mathcal{R}C\,;s$. By property (1) of DNFs, there is some $D \in \mathcal{D}(\mathcal{R}C\,;s)$ such that $\mathcal{I}$ satisfies D. The claim follows.
2. $s = \Diamond^* t$. Then $\mathcal{I}$ satisfies $\mathcal{R}C\,;t$ or $\mathcal{I}$ satisfies $\mathcal{R}C\,;\Diamond s$. Hence, there is some $D \in \mathcal{D}(\mathcal{R}C\,;t) \cup \mathcal{D}(\mathcal{R}C\,;\Diamond s)$ such that $\mathcal{I}$ satisfies D. The claim follows. ∎

Theorem 4.5 (Evidence) Let T be a complete tableau and $C \in T$. If C is satisfiable, then there is an evident subtableau of T containing C.

Proof Let T and C be as required, and let $\mathcal{I}$ be a model of C. We define U such that:

1. $D \in U \;:\Longleftrightarrow\; D \in T$ and $\mathcal{I}$ satisfies D.
2. $D^s E \in U \;:\Longleftrightarrow\; D^s E \in T$ and $\{D, E\} \subseteq U$.

Clearly, U is a subtableau of T. Since $C \in T$ and $\mathcal{I}$ satisfies C, $C \in U$. It remains to show that U is evident.

First, we show that, for all $\Diamond s \in D \in U$, there is some E such that $D^{\Diamond s}E \in U$. Let $\Diamond s \in D \in U$. Then $D \in T$ and $\mathcal{I}$ satisfies D. By Proposition 7.1, there is a link $D^{\Diamond s}E$ such that $\mathcal{I}$ satisfies E. Since T is complete, $D^{\Diamond s}E \in T$. The claim follows since $\mathcal{I}$ satisfies E.

We now show that U has a run for $D^{\Diamond^+ s}$ whenever $\Diamond^+ s \in D \in U$. Let $\Diamond^+ s \in D \in U$. Then $D \in T$ and $\mathcal{I}$ satisfies D. Hence, there are $X, Y, Z \in |\mathcal{I}|$ and some $n \geq 0$ such that $\mathcal{I}, X \vDash D$, $\mathcal{I}, Z \vDash s$, and $X \to_{\mathcal{I}} Y \to_{\mathcal{I}}^{n} Z$. By Proposition 3.4, $\mathcal{I}, Y \vDash \mathcal{R}D$. We proceed by induction on n.

Let $n = 0$. Then $Y = Z$, so $\mathcal{I}, Y \vDash \mathcal{R}D\,;s$. By property (1) of DNFs, there is some $E \in \mathcal{D}(\mathcal{R}D\,;s)$ such that $\mathcal{I}, Y \vDash E$. Since $E \in \mathcal{D}(\mathcal{R}D\,;s)$, $D^{\Diamond^+ s}E$ is a link, and hence $D^{\Diamond^+ s}E \in T$. Since $\mathcal{I}, Y \vDash E$, we have $D^{\Diamond^+ s}E \in U$. Since $E \rhd s$, DE is a run for $D^{\Diamond^+ s}$.

Let $n > 0$. Then $\mathcal{I}, Y \vDash \mathcal{R}D\,;\Diamond^+ s$. By property (1) of DNFs, there is some $E \in \mathcal{D}(\mathcal{R}D\,;\Diamond^+ s)$ such that $\mathcal{I}, Y \vDash E$. Since $E \in \mathcal{D}(\mathcal{R}D\,;\Diamond^+ s)$, $D^{\Diamond^+ s}E$ is a link, and hence $D^{\Diamond^+ s}E \in T$. Since $\mathcal{I}, Y \vDash E$, we have $D^{\Diamond^+ s}E \in U$. Since $E \rhd \Diamond^+ s$, $\Diamond^+ s \in E$. By the inductive hypothesis for $n - 1$, U has a run $E \ldots E'$ for $E^{\Diamond^+ s}$. Thus, $DE \ldots E'$ is a run for $D^{\Diamond^+ s}$ in U. ∎

The following propositions (7.2–7.5) can be shown for both $\mathrm{H_D}$ (using the definitions from § 5), and H^* and $\mathrm{H}^*_{\mathrm{D}}$ (using the corresponding definitions from § 6).

Proposition 7.2 If T is complete, $C, D \in T$, and $C^s D$ is a link, then $C^s D \in \hat{T}$.

Proposition 7.3 If $\mathcal{I}$ satisfies T and $\mathcal{I}, X \vDash C$, then $\mathcal{I}, X \vDash C^T$.

Proposition 7.4 If T is a complete tableau, $C \in T$, U is a subtableau of T, and C^U is a clause, then $C^U \in T$.

Proposition 7.1 adapts to nominals as follows:

Proposition 7.5 Let $\Diamond s \in C$ and let $\mathcal{I}$ satisfy C. Then there is a minimal link $C^{\Diamond s} D$ such that $\mathcal{I}$ satisfies D.

Theorem 5.5 (Evidence) Let T be a complete tableau and let $C \in T$ be such that, for all $t \in C$ and $\mathsf{D}s \in \mathcal{B}T$, $x_{\mathsf{D}s}$ does not occur in t. If C is satisfiable, then there is an evident subtableau U of $\hat{T}$ and a clause $D \in U$ such that $C \subseteq D$.

Proof Let T and C be as required, and let $\mathcal{I}$ be a model of C. Since no term in C contains nominals $x_{\mathsf{D}s}$ such that $\mathsf{D}s \in \mathcal{B}T$, without loss of generality we assume that, for all $\mathsf{D}s \in \mathcal{B}T$, $\mathcal{I}$ satisfies $\{x_{\mathsf{D}s},\ s\}$ whenever $\mathcal{I}$ satisfies $\{s\}$. We define U such that:

1. $D \in U \ :\Longleftrightarrow\ D \in T$, $\mathcal{I}$ satisfies D, and there is no $E \in T$ such that $\mathcal{I}$ satisfies E and $D \subsetneq E$.
2. $D^s E \in U \ :\Longleftrightarrow\ D^s E \in \hat{T}$ and $\{D, E\} \subseteq U$.

Clearly, U is a subtableau of $\hat{T}$. By assumption, we know that $\mathcal{I}$ satisfies C. Let D be a maximal clause in T such that $C \subseteq D$ and $\mathcal{I}$ satisfies D. By definition, $D \in U$. It remains to show that U is evident.

We begin with evidence condition (1). Let $\Diamond s \in C \in U$. Then $C \in T$ and $\mathcal{I}$ satisfies C. By Proposition 7.5, there is a minimal link $C^{\Diamond s} D$ such that $\mathcal{I}$ satisfies D. Since T is complete, $C^{\Diamond s} D \in T$. Since $\mathcal{I}$ satisfies D, there is some maximal $E \in T$ such that $D \subseteq E$ and $\mathcal{I}$ satisfies E. Then $E \in U$. Thus $C^{\Diamond s} E \in \hat{T}$, and hence $C^{\Diamond s} E \in U$.

Now to condition (2). Let $C \in U$ and $x \in \mathcal{B}C$. Then $C \in T$. Since T is complete, $\{x\} \in T$. Clearly, $\mathcal{I}, \mathcal{I}x \vDash \{x\}$. Hence, there is some maximal $D \in T$ such that $x \in D$ and $\mathcal{I}$ satisfies D. The claim follows since $D \in U$.

Now to (3). Let $C \in U$. Then $C \in T$ and $\mathcal{I}$ satisfies C. By Proposition 7.4, $C^U \in T$. By Proposition 7.3, $\mathcal{I}$ satisfies C^U. Clearly, we either have $C = C^U$ or $C \subsetneq C^U$. The claim follows since, by definition, C is a maximal clause in U that is satisfied by $\mathcal{I}$.

Now to condition (4). Let $\mathsf{D}s \in C \in U$. Then $C \in T$ and $\mathcal{I}$ satisfies C. We distinguish two cases:

- $x_{\mathsf{D}s} \notin C$. Then $\mathcal{I}$ satisfies $\{s\}$ and hence $\{x_{\mathsf{D}s},\ s\}$. Consequently, there is some $D \in \mathcal{D}\{x_{\mathsf{D}s},\ s\}$ such that $\mathcal{I}$ satisfies D. Since T is complete, $D \in T$. Let E be a maximal clause in T such that $D \subseteq E$ and $\mathcal{I}$ satisfies E. Then $E \in U$. By Proposition 3.1, $E \triangleright s$. Since $x_{\mathsf{D}s} \in E$, $E \neq C$.
- $x_{\mathsf{D}s} \in C$. Let X be a state such that $\mathcal{I}, X \vDash C$. Since $x_{\mathsf{D}s} \in C$, $\mathcal{I}x_{\mathsf{D}s} = X$. Since $\mathcal{I}$ satisfies C, there is some $Y \neq X$ such that $\mathcal{I}, Y \vDash s$. Since $Y \neq X$, $\mathcal{I}, Y \vDash \{\neg x_{\mathsf{D}s},\ s\}$. Hence, there is some $D \in \mathcal{D}\{\neg x_{\mathsf{D}s},\ s\}$ such that $\mathcal{I}$ satisfies D. The argument proceeds analogously to the first case.

Finally, to (5). Let $\bar{\mathsf{D}}s \in C \in U$, and let $D \in U$ such that $D \neq C$. We have to show that $D \triangleright s$. Assume, by contradiction, $D \not\triangleright s$. Let X, Y be states such that $\mathcal{I}, X \vDash C$ and $\mathcal{I}, Y \vDash D$. Since $\mathcal{I}$ satisfies C, we distinguish two cases:

- $X = Y$. Then $\mathcal{I}$ satisfies $C \cup D$. Since T is complete, $C \cup D \in T$. Let E be a maximal clause in T such that $C \cup D \subseteq E$ and $\mathcal{I}$ satisfies E. Clearly, $C, D \subsetneq C \cup D \subseteq E$, which contradicts the definition of U.
- $\mathcal{I}, Y \vDash s$. Then there is some $E \in \mathcal{D}(D\,;s)$ such that $\mathcal{I}, Y \vDash E$. Since T is complete, $E \in T$. Let E' be a maximal clause in T such that $E \subseteq E'$ and $\mathcal{I}$ satisfies E'. By Proposition 3.1, $D \subsetneq E \subseteq E'$, which contradicts the definition of U. ■

Proposition 7.6 If $C = C^T$ and $D = D^T$, then $C \cup D = (C \cup D)^T$.

Proposition 7.7 Let $\bar{\mathsf{D}}s \in C$ and $\mathcal{I}$ be a model of C. Then exactly one of the following two statements is true:

1. $\forall X \in |\mathcal{I}|\colon\ \mathcal{I}, X \vDash s$
2. $\exists X \in |\mathcal{I}|\colon\ \mathcal{I}, X \nvDash s,\ \mathcal{I}, X \vDash C$, and $\forall Y \in |\mathcal{I}|\colon\ X \neq Y \implies \mathcal{I}, Y \vDash s$ and $\mathcal{I}, Y \nvDash \bar{\mathsf{D}}s$

Proof Clearly, (1) and (2) cannot both be true at the same time. Assume (1) does not hold. It suffices to prove (2). Let X and Y be states (possibly identical) such that $\mathcal{I}, X \nvDash s$ and $\mathcal{I}, Y \nvDash s$. Since $\mathcal{I}$ satisfies C, there is a state Z such that $\mathcal{I}, Z \vDash C$. It now suffices to show that $X = Y = Z$, which follows since $\mathcal{I}, Z \vDash \bar{\mathsf{D}}s$. ■

Lemma 6.3 Let T be a complete tableau and $\mathcal{I}$ an interpretation. Let

$$T' := \{\, C \in T \mid \mathcal{I} \text{ satisfies } C \,\}$$

Let $\bar{\mathsf{D}}s_1 \ldots \bar{\mathsf{D}}s_n$ be an injective enumeration of the set $\{\, \bar{\mathsf{D}}s \mid \bar{\mathsf{D}}s \in C \in T' \,\}$. Let $T'_0 := \{\, C \in T' \mid C = C^{T'} \,\}$. For all $i \in [1, n]$ we construct a set T'_i from T'_{i-1} as follows:

- If $\forall X \in |\mathcal{I}|\colon\ \mathcal{I}, X \vDash s_i$, then $T'_i := \{\, C \in T'_{i-1} \mid C \triangleright s_i \,\}$.
- Otherwise, $T'_i := \{\, C \in T'_{i-1} \mid C \triangleright s_i \,\};\bigcup\{\, C \in T'_{i-1} \mid \bar{\mathsf{D}}s_i \in C \,\}$.

Then, for all $i \in [1, n]$:

1. If $C \in T'_{i-1} \cap T'_i$, $D \in T'_{i-1}$, and $C \subseteq D$, then $D \in T'_i$.
2. If $C \in T'_{i-1}$ and $\mathcal{I}, X \models C$, then $C \subseteq D$ for some $D \in T'_i$ such that $\mathcal{I}, X \models D$.
3. $T'_i \subseteq T'_{i-1}$.
4. If $C \in T'_i$, then $C = C^{T'_i}$ (i.e., T'_i satisfies evidence condition (4)).
5. Let $j \in [1, i]$, $\bar{\mathsf{D}}s_j \in C \in T'_i$, and $D \in T'_i$ such that $D \neq C$. Then $D \rhd s_j$ (i.e., T'_i satisfies evidence condition (6) restricted to $\bar{\mathsf{D}}s_1, \ldots, \bar{\mathsf{D}}s_i$).

Proof

1. Let $C \in T'_{i-1} \cap T'_i$ and let $D \in T'_{i-1}$ be a superset of C. We consider two cases:

- $T'_i = \{ C \in T'_{i-1} \mid C \rhd s_i \}$. Since $C \in T'_i$, $C \rhd s_i$. Since $D \supseteq C$, $D \rhd s_i$ (Proposition 3.1), and so $D \in T'_i$.
- $T'_i = \{ C \in T'_{i-1} \mid C \rhd s_i \} ; \bigcup \{ C \in T'_{i-1} \mid \bar{\mathsf{D}}s_i \in C \}$. Since $C \in T'_i$, we either have $C \rhd s_i$ or $\bar{\mathsf{D}}s_i \in C$. In the former subcase the proof proceeds analogously to the preceding case. In the latter subcase, we must have $C = \bigcup \{ C \in T'_{i-1} \mid \bar{\mathsf{D}}s_i \in C \}$. Consequently, since $\bar{\mathsf{D}}s_i \in C \subseteq D$, we have $C = D$. The claim follows.

2. The proof proceeds by induction on $i \in [1, n]$. Let $i \in [1, n]$ and let $C \in T'_{i-1}$. Without loss of generality let $C \not\rhd s_i$ (otherwise $C \in T'_i$). We distinguish two cases:

- $\forall X \in |\mathcal{I}|$: $\mathcal{I}, X \models s_i$. Let X be such that $\mathcal{I}, X \models C$. Then $\mathcal{I}, X \models C ; s_i$, so there is some $D \in \mathcal{D}(C ; s_i)$ such that $\mathcal{I}, X \models D$. Hence $D \in T'$, $D^{T'} \in T'_0$, and $\mathcal{I}, X \models D^{T'}$ (Propositions 7.3 and 7.4). We now show that there is some $E \in T'_{i-1}$ such that $C \subseteq D \subseteq D^{T'} \subseteq E$ and $\mathcal{I}, X \models E$. The claim then follows since $E \rhd s_i$, and so $E \in T'_i$. If $i = 1$, we can set $E = D^{T'}$. For $i > 1$, the existence of E follows by repeated application of the inductive hypothesis for $j \in [1, i-1]$.
- Otherwise, let X be a state such that $\mathcal{I}, X \not\models s_i$. Let Y be the state such that $\mathcal{I}, Y \models C$. If $Y \neq X$, $\mathcal{I}, Y \models s_i$ and the proof proceeds as in the first case. Otherwise, let $D \in T'$ be some clause such that $\bar{\mathsf{D}}s_i \in D$. Then $\mathcal{I}, Y \models C \cup D$ (Proposition 7.7), and hence $\mathcal{I}, Y \models (C \cup D)^{T'}$ and $(C \cup D)^{T'} \in T'_0$ (Propositions 7.3 and 7.4). As before, it now suffices to find some $E \in T'_{i-1}$ such that $C \subseteq (C \cup D)^{T'} \subseteq E$ and $\mathcal{I}, Y \models E$. The claim then follows since $\bar{\mathsf{D}}s_i \in E$, and so $E \subseteq \bigcup \{ C \in T'_{i-1} \mid \bar{\mathsf{D}}s_i \in C \}$. If $i = 1$, we can set $E = (C \cup D)^{T'}$, while for $i > 1$, the existence of E follows by repeated application of the inductive hypothesis for $j \in [1, i-1]$.

3. We proceed by induction on $i \in [1, n]$. Let $i \in [1, n]$. The claim is obvious if $T'_i = \{ C \in T'_{i-1} \mid C \rhd s_i \}$. Otherwise, it suffices to show that $\bigcup \{ C \in T'_{i-1} \mid \bar{\mathsf{D}}s_i \in C \} \in T'_{i-1}$. Let $D := \bigcup \{ C \in T'_{i-1} \mid \bar{\mathsf{D}}s_i \in C \}$. By Proposition 7.7, there is a state X such that

a) X is the unique state such that $\mathcal{I}, X \nvDash s_i$, and

b) X is the unique state such that $\mathcal{I}, X \vDash \bar{\mathsf{D}}s_i$.

By (b), D is a clause. By (a) and (b), we have $C \not\triangleright s_i$ whenever $\bar{\mathsf{D}}s_i \in C$. We have $D \in T$ by completeness criterion (6) and repeated application of the inductive hypothesis for $j \in [1, i-1]$ together with the observation that $T'_0 \subseteq T' \subseteq T$. Since $\mathcal{I}$ satisfies D, $D \in T'$. By construction, $C = C^{T'}$ holds for all $C \in T'_0$, and so, by repeated application of the inductive hypothesis, for all $C \in T'_{i-1}$. Then, by Proposition 7.6, $D = D^{T'}$, and hence $D \in T'_0$. Let $C \in T'_{i-1}$ such that $\bar{\mathsf{D}}s_i \in C$ (such a C exists by (2) and Propositions 7.3 and 7.4 since there is some $E \in T'$ such that $\bar{\mathsf{D}}s_i \in E$). By repeated application of the inductive hypothesis, $C \in T'_j$ for all $j \in [0, i-1]$. Since $C \subseteq D$, we obtain $D \in T'_{i-1}$ by repeated application of (1).

4. The claim holds since it holds for T'_0 and, by (3), $C^{T'_i} \subseteq C^{T'_{i-1}}$ for all $i \in [1, n]$.

5. We proceed by induction on $i \in [1, n]$. Let $i \in [1, n]$. Clearly, T'_i satisfies evidence condition (6) restricted to $\bar{\mathsf{D}}s_i$ if $T'_i = \{\, C \in T'_{i-1} \mid C \triangleright s_i \,\}$. Otherwise, we know that $C \not\triangleright s_i$ whenever $\bar{\mathsf{D}}s_i \in C$ (Proposition 7.7). Hence, by construction, for all clauses $C \in T'_i$ except $\bigcup\{\, C \in T'_{i-1} \mid \bar{\mathsf{D}}s_i \in C \,\}$ we have $C \triangleright s_i$ and $\bar{\mathsf{D}}s_i \notin C$. This implies evidence condition (6) restricted to $\bar{\mathsf{D}}s_i$. Evidence condition (6) restricted to $\bar{\mathsf{D}}s_1, \dots, \bar{\mathsf{D}}s_{i-1}$ follows by the inductive hypothesis and (3). ■

Theorem 6.4 (Evidence) Let T be a complete tableau and let $C \in T$ be such that, for all $t \in C$ and $\mathsf{D}s \in \mathcal{B}T$, $x_{\mathsf{D}s}$ does not occur in t. If C is satisfiable, then there is an evident subtableau U of $\hat{T}$ and a clause $D \in U$ such that $C \subseteq D$.

Proof Let T and C be as required, and let $\mathcal{I}$ be a model of C. Since no term in C contains nominals $x_{\mathsf{D}s}$ such that $\mathsf{D}s \in A$, without loss of generality we assume that, for all $\mathsf{D}s \in A$, $\mathcal{I}$ satisfies $\{x_{\mathsf{D}s}, s\}$ whenever $\mathcal{I}$ satisfies $\{s\}$. Let T', $\mathsf{D}s_1 \dots \mathsf{D}s_n$, $T'_1, \dots, T'_n$ be defined from T and $\mathcal{I}$ as in Lemma 6.3. We define:

$$U := T'_n \cup \{\, D^s E \in \hat{T} \mid \{D,\ E\} \subseteq T'_n \,\}$$

By Lemma 6.3 (3), $T'_n \subseteq T' \subseteq T$. Hence U is a subtableau of $\hat{T}$. Since $\mathcal{I}$ satisfies C, $C \in T'$. By Propositions 7.3 and 7.4, $C^{T'} \in T'$, and hence $C^{T'} \in T'_0$. So, by Lemma 6.3 (2), there is some $D \in T'_n \subseteq U$ such that $C \subseteq C^{T'} \subseteq D$.

By Lemma 6.3 (4,5), it remains to show that U satisfies evidence conditions (1), (2), (3) and (5). We begin with condition (5). Let $\mathsf{D}s \in E \in U$. We distinguish two cases.

- $x_{\mathsf{D}s} \notin E$. By Lemma 6.3 (3), $E \in T$. Since $\mathcal{I}$ satisfies E, $\mathcal{I}$ satisfies $\{s\}$ and hence $\{x_{\mathsf{D}s},\ s\}$. Consequently, there is some $E' \in \mathcal{D}\{x_{\mathsf{D}s},\ s\}$ such that $\mathcal{I}$ satisfies E'. Since T is complete, $E' \in T$. Since $\mathcal{I}$ satisfies E', $E' \in T'$. By Propositions 7.3, 7.4, and Lemma 6.3 (2), there is some $E'' \in U$ such that $E' \subseteq E'^{T'} \subseteq E''$. Clearly, $E'' \triangleright s$ (Proposition 3.1). Since $x_{\mathsf{D}s} \in E''$, $E'' \neq E$.

- $x_{\mathsf{D}s} \in E$. By Lemma 6.3 (3), $E \in T$. Let X be a state such that $\mathcal{I}, X \vDash E$. Since, $x_{\mathsf{D}s} \in E$, $\mathcal{I}x_{\mathsf{D}s} = X$. Since $\mathcal{I}$ satisfies E, there is some $Y \neq X$ such that $\mathcal{I}, Y \vDash s$. Since $Y \neq X$, $\mathcal{I}, Y \vDash \{\neg x_{\mathsf{D}s},\ s\}$. Hence, there is some $E' \in \mathcal{D}\{\neg x_{\mathsf{D}s},\ s\}$ such that $\mathcal{I}$ satisfies E'. The argument proceeds analogously to the first case.

Now to condition (1). Let $\Diamond s \in E \in U$. By Lemma 6.3 (3), $E \in T$. Since $\mathcal{I}$ satisfies E, by Proposition 7.5 there is a minimal link $E^{\Diamond s}E'$ such that $\mathcal{I}$ satisfies E'. Since T is complete, $E^{\Diamond s}E' \in T$, and hence $E' \in T$. By Propositions 7.3, 7.4, and Lemma 6.3 (2), there is some $E'' \in U$ such that $E' \subseteq E''$. Then $E^{\Diamond s}E''$ is a link and, by Lemma 6.3 (3), $E'' \in T$. By Proposition 7.2, $E^{\Diamond s}E'' \in \hat{T}$. The claim follows.

Now to (2). Let $\Diamond^+ s \in E \in U$. We show that U has a run for $E^{\Diamond^+ s}$. By Lemma 6.3 (3), $E \in T$. Since $\mathcal{I}$ satisfies E, there are states X, Y, Z and some $n \geq 0$ such that $\mathcal{I}, X \vDash E$, $\mathcal{I}, Z \vDash s$, and $X \to_{\mathcal{I}} Y \to_{\mathcal{I}}^{n} Z$. By Proposition 3.4, $\mathcal{I}, Y \vDash \mathcal{R}E$. We proceed by induction on n.

Let $n = 0$. Then $Y = Z$, so $\mathcal{I}, Y \vDash \mathcal{R}E\,;s$. Then there is some $E' \in \mathcal{D}(\mathcal{R}E\,;s)$ such that $\mathcal{I}, Y \vDash E'$. So, $E^{\Diamond^+ s}E'$ is a minimal link. Since T is complete, $E^{\Diamond^+ s}E' \in T$, and hence $E' \in T$. By Proposition 7.4 and Lemma 6.3 (2), there is some $E'' \in U$ such that $E' \subseteq E''$ (and $\mathcal{I}, Y \vDash E''$). Then $E^{\Diamond^+ s}E''$ is a link and, by Lemma 6.3 (3), $E'' \in T$. By Proposition 7.2, $E^{\Diamond^+ s}E'' \in \hat{T}$, and so $E^{\Diamond^+ s}E'' \in U$. Since $E'' \rhd s$, EE'' is a run for $E^{\Diamond^+ s}$.

Let $n > 0$. Then $\mathcal{I}, Y \vDash \mathcal{R}E\,;\Diamond^+ s$. Consequently, there is some $E' \in \mathcal{D}(\mathcal{R}E\,;\Diamond^+ s)$ such that $\mathcal{I}, Y \vDash E'$. Analogously to the case $n = 0$, we obtain that there is some $E'' \in U$ such that $E' \subseteq E''$, $\mathcal{I}, Y \vDash E''$, and $E^{\Diamond^+ s}E'' \in U$. Since $E'' \rhd \Diamond^+ s$, $\Diamond^+ s \in E''$. By the inductive hypothesis for $n-1$, U has a run $E'' \ldots F$ for $E''^{\Diamond^+ s}$. Thus $EE'' \ldots F$ is a run for $E^{\Diamond^+ s}$ in U.

It remains to show evidence condition (3). Let $E \in U$ and $x \in \mathcal{B}E$. By Lemma 6.3 (3), $E \in T$. Since T is complete, $\{x\} \in T$. Clearly, $\mathcal{I}, \mathcal{I}x \vDash \{x\}$. Hence, by Propositions 7.3, 7.4, and Lemma 6.3 (2), there is some $E' \in U$ such that $\{x\} \subseteq \{x\}^{T'} \subseteq E'$. The claim follows. ∎