The Computational Complexity of Hybrid Temporal Logics

CARLOS ARECES, *ILLC, Universiteit van Amsterdam, Plantage Muidergracht 24, 1018TV, Amsterdam, The Netherlands.*
E-mail: carlos@wins.uva.nl

PATRICK BLACKBURN, *Computerlinguistik, Universität des Saarlandes, 66041, Saarbrücken, Germany.*
E-mail: patrick@coli.uni-sb.de

MAARTEN MARX, *CCSOM, ILLC, Universiteit van Amsterdam, Plantage Muidergracht 24, 1018TV, Amsterdam, The Netherlands.*
E-mail: marx@wins.uva.nl

Abstract

In their simplest form, hybrid languages are propositional modal languages which can refer to states. They were introduced by Arthur Prior, the inventor of tense logic, and played an important role in his work: because they make reference to specific times possible, they remove the most serious obstacle to developing modal approaches to temporal representation and reasoning. However very little is known about the computational complexity of hybrid temporal logics.

In this paper we analyze the complexity of the satisfiability problem of a number of hybrid temporal logics: the *basic hybrid language* over transitive frames; *nominal tense logic* over transitive frames, strict total orders, and transitive trees; *nominal* Until *logic*; and *referential interval logic*. We discuss the effects of including nominals, the @ operator, the somewhere modality E, and the difference operator D. Adding nominals to tense logic leads for several frame–classes to an increase in complexity of the satisfiability problem from PSPACE to EXPTIME. On transitive trees, however, the satisfiability problem for this language can be decided in PSPACE. Along the way we make a detour through *hybrid* propositional dynamic logic: we establish upper bounds for a number of temporal logics by generalizing results due to Passy and Tinchev [PT91] and De Giacomo [De 95]. We conclude with some remarks on the relevance of our results to description logic, and draw attention to the utility of the *spypoint technique* for proving upper and lower bounds.

Keywords: Hybrid Logic, Computational Complexity, Modal and Temporal Logic, Propositional Dynamic Logic, Description Logic, Nominals, At Operator, Difference Operator, Universal Modality

1 Introduction

The simplest hybrid language is a uni-modal language that uses special atomic formulas (called *nominals*) to name states. Nominals are true at exactly one state in any model; they "name" this state by being true there and nowhere else. A wide range of hybrid languages have been studied, including hybrid languages in which it is possible to bind nominals in various ways, but although a number of fundamental complexity and undecidability results are known (see [ABM99]), the complexity of hybrid formalisms based on richer modal logics have rarely been studied. Two important exceptions to this are the EXPTIME-completeness result for Propositional Dynamic Logic (PDL) enriched with nominals and the somewhere modality E (a Sofia school result from the 1980s; see [PT91]), and the EXPTIME-completeness result for PDL with converse enriched with nominals (but without E) due to De Giacomo (see [De 95]). Moreover, most known results deal with the satisfiability problem over *arbitrary* frames; the only exception seems to be the results for deterministic frames proved in [BS93]. This is a real gap, as in practice we are often interested in richer modal languages and frames with certain properties. In particular, in temporal logic, we almost always work with transitive frames, and it is common to demand that temporal frames be treelike or even linear. This paper provides complexity results for various hybrid temporal logics over such structures.

Before going further, let's be precise about the syntax and semantics of the *basic hybrid language*, the weakest language we shall consider in the paper:

DEFINITION 1.1 (Syntax)
Let $\mathsf{PROP} = \{p, q, r, \ldots\}$ be a countable set of *propositional variables* and $\mathsf{NOM} = \{i, j, k, \ldots\}$ a countable set of *nominals*, disjoint from PROP. We call $\mathsf{ATOM} = \mathsf{PROP} \cup \mathsf{NOM}$ the set of *atoms*. The *well-formed formulas* of the basic hybrid language (over ATOM) are

$$\varphi \;:=\; a \mid \neg\varphi \mid \varphi \wedge \varphi' \mid \Diamond\varphi \mid\; @_i\varphi$$

where $a \in \mathsf{ATOM}$, and $i \in \mathsf{NOM}$. As usual, $\Box\varphi := \neg\Diamond\neg\varphi$.

Thus, syntactically speaking, the basic hybrid language is a two-sorted extension of uni-modal logic which contains a NOM indexed collection of operators $@_i$. Now for the semantics. Ordinary (unsorted) modal languages are interpreted on Kripke models, but hybrid languages are interpreted on *hybrid models*. These are Kripke models in which the valuations are constrained to ensure that each nominal names a unique state.

DEFINITION 1.2 (Semantics)
A *hybrid model* is a triple $\mathfrak{M} = \langle M, R, V\rangle$ such that M is a non-empty set, R is a binary relation on M, and $V : \mathsf{ATOM} \to Pow(M)$ is such that for all $i \in \mathsf{NOM}$, $V(i)$ is a singleton subset of M. The elements of M are called *states*, R is the *transition relation*, and V is the *valuation*. A *frame* is a pair $\mathfrak{F} = \langle M, R\rangle$, that is, a model without a valuation.

Let $\mathfrak{M} = \langle M, R, V\rangle$ be a model and $m \in M$. Then the *satisfaction relation* is defined by:

$$\begin{array}{lll}
\mathfrak{M}, m \Vdash a & \text{iff} & m \in V(a),\ a \in \mathsf{ATOM} \\
\mathfrak{M}, m \Vdash \neg\varphi & \text{iff} & \mathfrak{M}, m \nVdash \varphi \\
\mathfrak{M}, m \Vdash \varphi \wedge \psi & \text{iff} & \mathfrak{M}, m \Vdash \varphi \text{ and } \mathfrak{M}, m \Vdash \psi \\
\mathfrak{M}, m \Vdash \Diamond\varphi & \text{iff} & \exists m' \in M(Rmm' \;\&\; \mathfrak{M}, m' \Vdash \varphi) \\
\mathfrak{M}, m \Vdash @_i\varphi & \text{iff} & \mathfrak{M}, m' \Vdash \varphi, \text{ where } V(i) = \{m'\},\ i \in \mathsf{NOM}.
\end{array}$$

A formula φ is *satisfiable* if there is a hybrid model $\mathfrak{M}$, and a state $m \in M$ such that $\mathfrak{M}, m \Vdash \varphi$. We write $\mathfrak{M} \Vdash \varphi$ if for all $m \in M$, $\mathfrak{M}, m \Vdash \varphi$, and say that φ is *globally* satisfied in $\mathfrak{M}$. If $\mathfrak{F}$ is a frame, and for all hybrid valuations V on $\mathfrak{F}$ we have $\langle \mathfrak{F}, V\rangle \Vdash \varphi$, we say that φ is *valid* on $\mathfrak{F}$ and write $\mathfrak{F} \Vdash \varphi$.

Because hybrid valuations assign *singletons* to nominals, each nominal is satisfied at exactly one state in any model; this means that nominals are essentially a mechanism for referring to states in a propositional modal language. In what follows we often call the unique state in $V(i)$ the *denotation* of i. Note that the satisfaction clause for formulas of the form $@_i\varphi$ says: to evaluate $@_i\varphi$, jump to the unique state named by i and evaluate φ there. It follows that a wff of the form $@_ij$ expresses the identity of the states named by i and j, and that a wff of the form $@_i\Diamond j$ says that the state named by i R-precedes the state named by j.

So far we have said nothing about why we are interested in hybrid languages in the first place. There are at least two good reasons. First, they are proof-theoretically well behaved. Second, hybrid languages arise naturally in many applications.

Hybrid Languages and Modal Proof Theory Probably the most straightforward way to see why hybrid languages are proof-theoretically natural is to observe that the basic hybrid language offers precisely what is needed to capture the main ideas of labelled deduction. Labelled deduction (see [Gab96]) is built around the notation $l:\varphi$. Here the metalinguistic symbol : associates the metalinguistic label l with the object language formula φ. This has a natural modal interpretation: regard labels as names for states and read $l:\varphi$ as asserting that φ is satisfied at l. Labelled deduction proceeds by manipulating such labels to guide proof search, and is today an important approach to modal proof theory (see, for example, the work of Basin, Matthews and Viganò [BMV97]).

The basic hybrid language places the apparatus of labelled deduction in the *object* language: nominals are essentially object-level labels, and the formula $@_i\varphi$ asserts in the object language what $i : \varphi$ asserts in the metalanguage. And indeed, the basic hybrid language turns out to be proof-theoretically natural, for we can directly "internalize" standard labelled deduction (see [Sel97, Bla98]). Nonetheless, though important, @ is not indispensable. For example, [Tza99] presents a general approach to hybrid proof theory using Fitting-style prefix calculi. And when the underlying modal logic is tense logic (that is, when we have a backward looking modality to match the forward looking one) even more flexibility is possible: [Dem99] presents a sequent system for nominal tense logic *without* @ that has much in common with @-based internalized labelled deductive systems, and [DG99] discusses display calculi for nominal tense logic with the difference operator D. For this reason, when we discuss nominal tense logic, we will be interested in the complexity of @-free systems, system containing @, and even richer systems containing the universal modality E or the difference modality D.

Hybrid Languages and Applied Logic Modal logicians like to point out that many of the formalisms invented by researchers in artificial intelligence, computational linguistics, and other fields are actually notational variants of modal logic. But in many cases it would be more accurate to say that it is *hybrid languages* which are reinvented in this way. For example, it is well known that the description language $\mathcal{ALC}$ (see [SSS91]) is a notational variant of multi-modal logic (see [Sch91]). But this relation only holds at the level of T-Box reasoning (that is, general reasoning about concepts). It is often important to perform A-Box (or assertional) reasoning as well — that is, to reason about how concepts apply to particular individuals. A-Box reasoning corresponds to a restricted use of nominals and the $@_i$ operators (see [BT98, AdR99]). In a similar vein, while the basic Attribute Value Matrices (AVMs) used in computational linguistics are an obvious notational variant of deterministic multi-modal logic, full AVM notation (which provides a special device for specifying that two sequences of features lead to exactly the same node) corresponds to deterministic multi-modal logic enriched with nominals (see [BS93] and [Rea94]).

But in fact hybrid languages were introduced in the mid-1960s: they were developed by Arthur Prior, the founder of tense logic, and played a crucial role in his modal analysis of temporal representation and reasoning (see [Pri67], especially Chapter 5 and Appendix B3). Since then, many different hybrid temporal logics have been explored for different purposes (see [Bul70, Has91, Bla94, Gor96, BT99]). Among other things, hybrid languages make it possible to introduce specific times (days, dates, years, etc.), to cope with temporal indexicals (such as yesterday, today, tomorrow and now) and to define many temporally relevant frame properties (such as irreflexivity, asymmetry, trichotomy, and directedness) that ordinary modal languages cannot. Furthermore, as we discuss in Section 5, when nominals and @ are added to interval-based logic, the result is a $\mathsf{Holds}(t, \varphi)$-driven interval logic in the style of those introduced into AI by James Allen, with @ playing the role of Holds. In a nutshell, because they make temporal reference possible, nominals (aided, perhaps, by @, E or D) remove the most serious obstacle to a modal analysis of temporal representation and reasoning.

Unfortunately, very little is known about the complexity of hybrid logics, and in particular of hybrid temporal logics. This paper is a first attempt to fill the gap. We will concentrate on complexity results for nominal tense logic, presenting a series of results which show that the EXPTIME-completeness of nominal tense logic reported in [ABM99] can be tamed by either choosing a more appropriate class of frames, or by working with a fragment of the language.

Overview of the paper. In the next section we discuss the relations between nominals, the universal modality and the difference operator. Section 3 contains our main results. We first show that the satisfiability problem for the basic hybrid language is complete for PSPACE when interpreted on transitive frames. Then we show that adding just one nominal to the language of tense logic leads to EXPTIME hardness on this class of frames. We then consider frame classes which are more appropriate to temporal logic: linear structures and transitive trees. In both cases we establish that the complexity of the satisfiability problem remains the same after expansion with nominals and the @ operator. The last two sections discuss two stronger languages: nominal Until/Since logic, and a referential version of Halpern and Shoham's interval logic. Along the way we make a brief detour through hybrid PDL: we establish upper bounds for a number of temporal logics by generalizing results due to Passy and Tinchev [PT91] and De Giacomo [De 95]. We conclude with some remarks on the relevance of our results to description logic, and draw attention to the *spypoint technique*, which is used in this paper to prove both upper and lower bounds. We believe this technique may be useful in other settings.

2 Nominals, @, E and D

The somewhere modality E, and to a lesser extent the difference operator D, have played an important role in the development of hybrid languages. These operators can be added to any (modal or hybrid) language, and on any (Kripke or hybrid) model $\mathfrak{M} = \langle M, R, V\rangle$ they are interpreted as follows:

$$\begin{array}{lll}\mathfrak{M}, m \Vdash \mathsf{E}\varphi & \text{iff} & \exists m' \in M(\mathfrak{M}, m' \Vdash \varphi)\\ \mathfrak{M}, m \Vdash \mathsf{D}\varphi & \text{iff} & \exists m' \in M(m' \neq m \ \wedge\ \mathfrak{M}, m' \Vdash \varphi).\end{array}$$

We will see a lot more of these operators in what follows, so let's note some useful facts right away. First, the dual of E, namely $\mathsf{A}\varphi := \neg\mathsf{E}\neg\varphi$, asserts that φ is true at *all* points in a model; A is called the *universal modality*. Second, note that E can mimic @, for $\mathsf{E}(i \wedge \varphi)$ means exactly the same thing as $@_i\varphi$; this is why practically everyone who has worked with nominals has also experimented with E. In a sense, @ amounts to *guarded* use of E, where "guarded" is used in the sense of [AvBN98]. As we shall see, this kind of guarding can be effective: hybrid logics with @ are often less complex than those which allow unrestricted use of E.

The difference operator D is stronger than E, for we can define $\mathsf{E}\varphi$ as $\varphi \vee \mathsf{D}\varphi$ but E is not strong enough to define D. Actually, D is so strong that it can even simulate nominals: clearly the formula $\mathsf{E}p \wedge \mathsf{A}(p \to \neg\mathsf{D}p)$ forces p to be true at only one point in the model.[1] More interesting is the following fact:

Theorem 2.1
There is a polynomial reduction which preserves satisfaction from any hybrid language containing D to the fragment containing only E and nominals.

This result is relevant to our work on hybrid complexity, for it immediately yields:

Corollary 2.2
Let F be any class of Kripke frames, and let $\mathcal{L}$ be a modal language. The F-satisfiability problem is the problem of satisfying a formula on a model based on a frame in F. Then:

1. if $\mathcal{L}$ contains D then adding nominals, @, and E does not modify (except by a polynomial) the complexity of the F-satisfiability problem,
2. if $\mathcal{L}$ contains nominals and E then adding D does not modify (except by a polynomial) the complexity of the F-satisfiability problem.

For this reason we won't explicitly mention results concerning D. We now prove the theorem.

Proof. Let φ be a formula in the full language. We construct in two steps a formula $\varphi' \wedge \theta'$ without D such that φ is satisfiable iff $\varphi' \wedge \theta'$ is. We take care that this construction can be performed in time polynomial in the length of φ. We use the fact that in any Kripke model $\mathfrak{M}$, the denotation of $\mathsf{D}\varphi$ (denoted by $[\mathsf{D}\varphi]_{\mathfrak{M}} = \{m \in M \mid \mathfrak{M}, m \Vdash \mathsf{D}\varphi\}$) can only take three values, namely:

$$[\mathsf{D}\varphi]_{\mathfrak{M}} = \begin{cases} M & \text{if } |[\varphi]_{\mathfrak{M}}| > 1\\ \emptyset & \text{if } [\varphi]_{\mathfrak{M}} = \emptyset\\ M \setminus \{m\} & \text{if } [\varphi]_{\mathfrak{M}} = \{m\}.\end{cases}$$

We now delete all occurrences of D, replacing them with nominals and A; we proceed inductively in the number of D operators in φ. If φ contains no D we are done. Otherwise, consider a subformula of the form $\mathsf{D}\psi$ where ψ contains no occurrences

[1] This is not a new observation. For example, proof systems for D-logics based on rules for the undefinable trade on this; see [Rij92, Ven93].

of D. Let φ' be φ with this subformula replaced by a new variable p_k and let $\theta_k = \mathsf{A}(p_k \leftrightarrow \mathsf{D}\psi)$. Clearly φ is satisfiable iff $\varphi' \wedge \theta_k$ is. An inductive application of this procedure eventually yields a formula φ' without occurrences of D and a conjunction θ of formulas of the form $\mathsf{A}(p_k \leftrightarrow \mathsf{D}\psi)$ with ψ not containing D.

Now we "axiomatize" all the $\mathsf{D}\psi$ using nominals, E, and A. For every conjunct $\theta_k = \mathsf{A}(p_k \leftrightarrow \mathsf{D}\psi)$, we create a formula θ'_k, the conjunction of:

$$\begin{array}{l} \mathsf{A}p_k \vee \mathsf{A}\neg p_k \vee (\mathsf{A}(p_k \leftrightarrow \neg i_k) \wedge \mathsf{E}p_k) \\ \mathsf{A}p_k \to \mathsf{E}(\psi \wedge i_k) \wedge \mathsf{E}(\psi \wedge \neg i_k) \\ \mathsf{A}\neg p_k \to \mathsf{A}\neg\psi \\ (\mathsf{A}(p_k \leftrightarrow \neg i_k) \wedge \mathsf{E}p_k) \to \mathsf{A}(\psi \to i_k). \end{array}$$

For every conjunct θ_k, we use a new nominal i_k. Note that the three disjuncts making up the first conjunct are mutually exclusive: each reflects one of the three possible denotations of $\mathsf{D}\psi$. Let θ' be the conjunction of all θ'_k. It is clear that $\varphi' \wedge \theta$ is satisfiable iff $\varphi' \wedge \theta'$ is, and θ' contains no occurrences of D. Finally, the translation is a PTIME-reduction with at most a quadratic blow-up in the size of the formula. ∎

3 Nominal tense logic

The classic language for temporal reasoning is Priorean tense logic [Pri67]. This is a bimodal language whose $\Diamond$-modalities are F (for reasoning about the *future*) and P (for reasoning about the *past*). P is interpreted using the converse of the transition relation for F. That is:

$$\begin{array}{lll} \mathfrak{M}, m \Vdash \mathsf{F}\varphi & \text{iff} & \exists m' \in M(Rmm' \;\&\; \mathfrak{M}, m' \Vdash \varphi) \\ \mathfrak{M}, m \Vdash \mathsf{P}\varphi & \text{iff} & \exists m' \in M(Rm'm \;\&\; \mathfrak{M}, m' \Vdash \varphi). \end{array}$$

The respective $\Box$-modalities are written G and H. We know from [Spa93b] that the satisfiability problem for Priorean tense logic over arbitrary frames is PSPACE-complete, and that it remains PSPACE-complete over transitive frames.

When nominals (and possibly @) are added to tense logic, the resulting language is called *nominal tense logic*. Nominal tense logic is well-behaved in many respects, and its good behavior does not depend on @. For example, [Dem99] shows how to define well-behaved @-free sequent systems. Moreover, nominal tense logic is highly expressive at the level of frames. For example, it can define both $\langle \mathbb{N}, < \rangle$ (the natural numbers in their usual order) and $\langle \mathbb{Z}, < \rangle$ (the integers in their usual order) up to isomorphism (see [Bla93]) and the definitions don't require @. In what follows, when stating complexity results we will explicitly say whether they presuppose the presence of @ or not.

What about complexity? The following result, proved in [ABM99], is a good starting point as we can consider the basic hybrid language as the purely future fragment of nominal tense logic.

THEOREM 3.1
The satisfiability problem for the basic hybrid language over arbitrary frames is PSPACE-complete.

Now, we know from [Lad77] that ordinary propositional uni-modal logic has a PSPACE-complete satisfiability problem over arbitrary frames. Hence, Theorem 3.1 shows that we can strengthen the expressivity of the logic by adding nominals and @ without any complexity cost (up to a polynomial). Furthermore, it shows that for arbitrary frames, the guarding strategy (that is, using @ instead of E) pays off: adding the somewhere modality E (and no nominals) to ordinary uni-modal logic results in an EXPTIME–hard satisfiability problem (see [FL79, HM92]).

But these results are of limited *temporal* interest for two reasons. First, while the purely future fragment is useful in certain applications, we often want the past operator too. Moreover, if we think of states in a Kripke model as *time points*, and view R as the *temporal precedence* relation, reading $\Diamond\varphi$ as "*φ occurs in the future*," then this means we should require R to be (at least) *transitive*.

In [ABM99] it was shown that the past operator does make a difference. We pay a price for the ability to look backwards:

THEOREM 3.2
The satisfiability problem over arbitrary frames for the Priorean tense language expanded with just one nominal (and no @ operator) is EXPTIME–hard.

This sounds like bad new — but because the class of arbitrary frames is of little relevance in temporal logic, it is merely a warning sign. The real task that faces us is to examine the complexity of hybrid logics over the frame classes that really are important in temporal logic. We start by looking at the minimal requirement we can impose on R, transitivity.

3.1 Transitive frames.

We start again with the simple case of the pure future fragment (that is, the basic hybrid language). We know (also from [Lad77]) that the satisfiability problem for ordinary uni-modal logic over transitive frames is PSPACE-complete. What happens when we add nominals and @? Again, nothing. In fact we can even add the somewhere modality E while staying in PSPACE.

THEOREM 3.3
The satisfiability problem over transitive frames for the basic hybrid language expanded with E is PSPACE-complete.

PROOF. The lower bound follows from [Lad77]. For the proof of the upper bound, we only consider formulas without occurrences of @. We lose no generality because $@_i\varphi$ is equivalent to $\mathsf{E}(i \wedge \varphi)$.

For a formula ξ we will define a two player ξ-game. We will show that ∃loise (the existential player) has a winning strategy in the ξ-game if and only if ξ is satisfiable. The ξ-game is designed so that it halts after at most $|SF(\xi)|$ rounds, where $SF(\xi)$ is the set of all subformulas in ξ. Moreover, at each stage of the game at most $|SF(\xi)|$ Hintikka sets (maximal consistent sets of subformulas of ξ) are on the board. Coding this information takes space polynomial in the length of ξ. Given the close correspondence between Alternating Turing Machines (ATM's) and two player games (see [Chl86]), it is straightforward to implement the problem of deciding whether ∃loise has a winning strategy on a PTIME ATM. As any PTIME ATM algorithm can be turned into a PSPACE Turing Machine program, we obtain the desired upper bound.

Fix a formula ξ. The ξ-game is played as follows. There are two players, ∀belard (male) and ∃loise (female). She starts the game by playing a collection $\{X_0, \ldots, X_k\}$

of Hintikka sets. She must choose these Hintikka sets so that the following three conditions hold:

1. X_0 contains ξ, and $|\{X_0, \ldots, X_k\}|$ is smaller than $|SF(\xi)|$,
2. no nominal occurs in two different Hintikka sets,
3. for all X_l, for all $\mathsf{E}\varphi \in SF(\xi)$, $\mathsf{E}\varphi \in X_l$ iff $\varphi \in X_k$, for some k.

If $\exists$loise cannot find Hintikka sets satisfying these conditions, she loses the game immediately. If she can, the game continues. $\forall$belard chooses an X_l and a "defect formula" $\Diamond\varphi \in X_l$. $\exists$loise must respond with a Hintikka set Y such that

1. $\varphi \in Y$ and for all $\Diamond\psi \in SF(\xi)$, $\Diamond\psi \notin X_l$ implies that $\Diamond\psi \notin Y$ and $\psi \notin Y$,
2. for all $\mathsf{E}\varphi \in SF(\xi)$, if $\varphi \in Y$ then $\mathsf{E}\varphi \in Y$, and $\mathsf{E}\varphi \in Y$ iff $\mathsf{E}\varphi \in X_l$,
3. if $i \in Y$ for some nominal i, then Y is one of the Hintikka sets she played at the start. In this case the game stops and $\exists$loise wins.

If $\exists$loise cannot find a suitable Y, the game stops and $\forall$belard wins. If $\exists$loise does find a suitable Y (one that is not covered by the halting clause in item 3 above) then Y is added to the list of played sets, and play continues, the players moving alternately. At each stage, $\forall$belard must choose a defect $\Diamond\varphi$ from the last played Hintikka set. To ensure that the the length of the game is bounded by $|SF(\xi)|$, we keep a list of the $\Diamond$-formulas $\forall$belard plays, and we insist that if he plays a formula $\Diamond\varphi$ a second time, $\exists$loise has to respond with the Hintikka set she played when he challenged with $\Diamond\varphi$ the first time. If this (forced) response does not meet the three criteria just listed, she loses; but if it does meet these criteria, she wins. Either way, the game stops immediately.

Claim: $\exists$loise has a winning strategy in the ξ-game iff ξ is satisfiable in the class of transitive hybrid models.

Proof of Claim.
[$\Rightarrow$]. Suppose $\exists$loise has a winning strategy for the ξ-game. We build a model $\mathfrak{M}$ for ξ as follows. The domain M is build in steps by following her winning strategy. M_0 consists of her initial move $\{X_0, \ldots, X_k\}$. Suppose M_j is defined. Then M_{j+1} consists of a copy of those Hintikka sets she plays when using her winning strategy for each of $\forall$belard's possible moves played in the Hintikka sets from M_j (except when she is required to play a Hintikka set from her initial move that contains a nominal; we don't copy these sets). Let M be the disjoint union of all the M_j. Set Rmm' iff for all $\Diamond\varphi \in SF(\xi)$, $\Diamond\varphi \notin m \Rightarrow [\Diamond\varphi \notin m' \;\&\; \varphi \notin m']$. Let $V(a) = \{m \in M \mid a \in m\}$, for all atoms a in $SF(\xi)$; and if i is any nominal *not* in $SF(\xi)$, then $V(i)$ is an arbitrary singleton subset of M. Clearly $\mathfrak{M}$ is a transitive hybrid model.

We claim that the following truth-lemma holds: for all $\psi \in SF(\xi)$, for all $m \in M$, $\mathfrak{M}, m \Vdash \psi$ if and only if $\psi \in m$. This follows by induction on the structure of formulas. The steps for atoms, Booleans and E are obvious. For $\Diamond$, first suppose that $\Diamond\psi \in m$. Then, as $\forall$belard can challenge this defect, it must be possible for $\exists$loise to respond with an m' containing ψ. Since for all $\Diamond\psi \in SF(\xi)$, $\Diamond\psi \notin m \Rightarrow [\Diamond\varphi \notin m' \;\&\; \varphi \notin m']$ holds, we have Rmm' and by induction hypothesis $\mathfrak{M}, m \Vdash \Diamond\psi$. For the converse, we argue by contraposition. Suppose that $\Diamond\psi \notin m$. Let m' be an arbitrary R-successor of m. Then by our definition of R, $\psi \notin m'$, so $\mathfrak{M}, m' \nVdash \psi$. As m' was arbitrary, $\mathfrak{M}, m \nVdash \Diamond\psi$ as required. Thus the truth-lemma holds, and as $\exists$loise played a Hintikka set containing ξ in the first round, ξ is satisfiable in $\mathfrak{M}$. We have proved the left to right direction of our claim.

[$\Leftarrow$]. Suppose ξ is satisfiable. That is, suppose there is some hybrid model $\mathfrak{M}$ and a point m_0 such that $\mathfrak{M}, m_0 \Vdash \xi$. Let $\mathfrak{M}^f = \langle M^f, R^f, V^f \rangle$ be the transitive filtration of $\mathfrak{M}$ through $SF(\xi)$.[2] Then $\mathfrak{M}^f$ is a *finite* model, $\mathfrak{M}^f, |m_0| \Vdash \xi$, and moreover each state in $\mathfrak{M}^f$ is a Hintikka set over ξ. Think of $\exists$loise as consulting $\mathfrak{M}^f$ as she plays, and choosing her moves from its states. For her first move, $\exists$loise chooses $|m_0|$ (as this is a point in $\mathfrak{M}^f$ that contains ξ), and each state of $\mathfrak{M}^f$ that contains a nominal and for every $\mathsf{E}\psi \in SF(\xi)$, if ψ is satisfied in $\mathfrak{M}^f$, just one $|m|$ such that $\mathfrak{M}^f, |m| \Vdash \psi$. Clearly her first move satisfies the required conditions. Now for the crucial point: when $\forall$belard chooses a defect $\Diamond\varphi$ from a Hintikka set X, $\exists$loise responds with a *maximal* R^f-successor Y of X such that Y contains φ; as $\mathfrak{M}^f$ is finite, such a Y exists. It is this choice that enables $\exists$loise to successfully play the same Hintikka set twice if $\forall$belard chooses a defect $\Diamond\varphi$ twice. For suppose $\exists$loise has played Y in response to a defect $\Diamond\varphi$ in X. Then suppose that at some later stage $\forall$belard points to the defect $\Diamond\varphi$ in Z, for Z a successor of X. $\exists$loise consults the model, looking for a maximal R^f successor of Z that contains φ. But as $R^f XZ$, any such point will also be a maximal R^f successor of X that contains φ. Thus Y is a suitable choice, and by playing it again she wins immediately. Thus $\exists$loise has a winning strategy for ξ-game. This completes the proof of the right to left direction of our claim. ◀

The theorem follows directly from the claim. ∎

So far, so good: nothing strange happens when we add transitivity to the basic hybrid language. But as we will now show, transitivity is not enough to tame hybridization when the backward looking modality P is added. The proof that follows uses the *spypoint technique* from [BS95, ABM99].

THEOREM 3.4
The satisfiability problem over transitive frames for the Priorean tense language expanded with just one nominal (and no @ operator) is EXPTIME–hard.

PROOF. We will reduce the EXPTIME-complete *global* K-satisfiability problem for uni-modal languages to the (local) satisfiability problem over transitive frames for a language of Priorean tense logic that contains at least one nominal (and no @ operator). The global K-satisfiability problem for uni-modal languages is this: given a formula φ in the uni-modal language, does there exist a Kripke model $\mathfrak{M}$ such that $\mathfrak{M} \Vdash \varphi$ (in other words, where φ is true in *all* states)? The EXPTIME-completeness of this problem is an immediate consequence of the EXPTIME-completeness of the satisfiability problem over arbitrary frames for uni-modal logic expanded with the somewhere modality E proved in [Spa93a].

We define a translation function $(\cdot)^t$ from ordinary uni-modal formulas to formulas in a language of tense logic containing one nominal i. We assume without loss of generality that the propositional variables in the nominal language are those of the uni-modal language plus four extra propositional variables 0, 1, 2, and 3. The translation $(\cdot)^t$ is defined as follows:

[2] $\mathfrak{M}^f$ is defined as follows. For each state $m \in \mathfrak{M}$, $|m|$ is the set of all states in $\mathfrak{M}$ that agree on all formulas in $SF(\xi)$, and $M^f = \{|m| \mid m \text{ is a state in } \mathfrak{M}\}$. R^f is the binary relation on M_f defined by $R^f|m||m'|$ iff for all ψ, if $\Diamond\psi \in SF(\xi)$ and $\mathfrak{M}, m \not\Vdash \Diamond\psi$, then $\mathfrak{M}, m' \not\Vdash \psi$ and $\mathfrak{M}, m' \not\Vdash \Diamond\psi$. $V^f(a) = \{|m| \mid \mathfrak{M}, w \Vdash a\}$, for all atomic symbols $a \in SF(\xi)$, and V^f assigns arbitrary singletons to nominals not in $SF(\xi)$.

$$\begin{array}{rcl} p^t & = & p \\ (\neg\xi)^t & = & \neg\xi^t \\ (\xi \wedge \theta)^t & = & \xi^t \wedge \theta^t \\ (\Diamond\xi)^t & = & \mathsf{F}(1 \wedge \mathsf{P}(2 \wedge \mathsf{F}(3 \wedge \mathsf{P}(0 \wedge \mathsf{P}i \wedge \xi^t)))). \end{array}$$

Clearly $(\cdot)^t$ is a linear reduction. The intuition behind this translation is that we mimic one R-step in an ordinary Kripke frame by the zigzag transition shown in Figure 1 in a transitive hybrid frame.

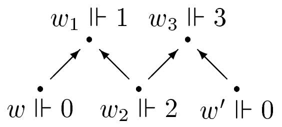


FIG. 1. Translation of diamond formulas.

The figure shows how an arrow from w to w' in the original model would be encoded in the transitive model, with intermediate stops at w_1, w_2 and w_3. The propositional symbol 0 will mark the elements of the original model, the others are auxiliary in the encoding. Note that on transitive frames, $\mathsf{FF}p \to \mathsf{F}p$ and $\mathsf{PP}p \to \mathsf{P}p$ are valid, but $\mathsf{FPFP}p \to \mathsf{FP}p$ is not.

CLAIM: For any uni-modal formula φ, φ is globally K-satisfiable iff $i \wedge \mathsf{F}0 \wedge \mathsf{G}(0 \to \varphi^t)$ is satisfiable in a hybrid model based on a transitive frame.

PROOF OF CLAIM.
[$\Rightarrow$]. Suppose $\mathfrak{M} \Vdash \varphi$, where $\mathfrak{M} = \langle M, R, V\rangle$ is a Kripke model. We now define a transitive hybrid model $\mathfrak{M}^{4s} = \langle M^{4s}, R^{4s}, V^{4s}\rangle$ as follows:

- $M^{4s} = (M \times \{\mathbf{0},\mathbf{1},\mathbf{2},\mathbf{3}\}) \cup \{s\}$, for $s \notin M \times \{\mathbf{0},\mathbf{1},\mathbf{2},\mathbf{3}\}$. For $x \in \{\mathbf{0},\mathbf{1},\mathbf{2},\mathbf{3}\}$, M^x is $\{(m,x) \mid m \in M\}$. In what follows, for all $m \in M$, we shall use the notation m_0, m_1, m_2, and m_3 instead of $(m,\mathbf{0})$, $(m,\mathbf{1})$, $(m,\mathbf{2})$, and $(m,\mathbf{3})$. Points in M^0 will correspond to the points in the Kripke model $\mathfrak{M}$; points in M^1, M^2 and M^3 are a "coding space" in which we can construct the transitive relation we require. The point s is our *spypoint*.
- R^{4s} is defined as follows

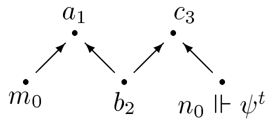
$$R^{4s}ab \iff \begin{cases} a = x_0, b = x_3 & \text{for some } x \in M, \text{ or} \\ a = x_2, b = x_3 & \text{for some } x \in M, \text{ or} \\ a = x_2, b = x_1 & \text{for some } x \in M, \text{ or} \\ a = y_0, b = x_1 & \text{for some } y, x \in M \ \& \ Ryx, \text{ or} \\ a = s & \end{cases}$$

  R^{4s} codes an R-transition from point y to x in $\mathfrak{M}$ as a forward step from y_0 to x_1 followed by a backward, forward, backward sequence from x_1 to x_3. Note that this zig-zag sequence corresponds to the nesting pattern of the Fs and Ps in the translation clause for $\Diamond$. We leave the reader to verify that R^{4s} is a *transitive* relation. Note that the spypoint can see every point in the model.
- If p is a propositional variable not in $\{1,2,3,4\}$, then $V^{4s}(p) = \{m_0 \mid m \in V(p)\}$; for $p \in \{1,2,3,4\}$, $V^{4s}(p) = M^p$, and $V^{4s}(i) = \{s\}$.

It follows by induction that for all $m \in M$, for all uni-modal formulas ψ, $\mathfrak{M}, m \Vdash \psi$ iff $\mathfrak{M}^{4s}, m_0 \Vdash \psi^t$. The only interesting step is for $\Diamond$. That $\mathfrak{M}, m \Vdash \Diamond\psi$ implies $\mathfrak{M}^{4s}, m_0 \Vdash (\Diamond\psi)^t$ is straightforward. For the reverse implication, suppose that $\mathfrak{M}^{4s}, m_0 \Vdash (\Diamond\psi)^t$, that is, suppose

$$\mathfrak{M}^{4s}, m_0 \Vdash \mathsf{F}(1 \wedge \mathsf{P}(2 \wedge \mathsf{F}(3 \wedge \mathsf{P}(0 \wedge \mathsf{P}i \wedge \psi^t)))).$$

This implies that there exist $a_1 \in M^1$, $b_2 \in M^2$, $c_3 \in M^3$ and $n_0 \in M^0$ such that



But by the definition of R^{4s} it follows that $a_1 = n_1$, $b_2 = n_2$, and $c_3 = n_3$. Hence, as $R^{4s}m_0a_1$, it follows that Rmn. Moreover, as $\mathfrak{M}^{4s}, n_0 \Vdash \psi^t$, by the inductive hypothesis $\mathfrak{M}, n \Vdash \psi$. Thus $\mathfrak{M}, m \Vdash \Diamond\psi$.

From this equivalence it follows that $\mathfrak{M}^{4s}, s \Vdash i \wedge \mathsf{F}0 \wedge \mathsf{G}(0 \to \varphi^t)$, and we have proved the left to right direction of our claim.

[$\Leftarrow$]. Suppose $\mathfrak{M}, w \Vdash i \wedge \mathsf{F}0 \wedge \mathsf{G}(0 \to \varphi^t)$, where $\mathfrak{M} = \langle M, R, V\rangle$ is a transitive hybrid model. Define a Kripke model $\mathfrak{M}^\blacktriangledown$ as follows:

- $M^\blacktriangledown = \{m \in M \mid Rwm \;\&\; \mathfrak{M}, m \Vdash 0\}$. Note that $M^\blacktriangledown$ is not empty, for $\mathfrak{M}, w \Vdash \mathsf{F}0$.
- $R^\blacktriangledown = \{(m,n) \in M^\blacktriangledown \times M^\blacktriangledown \mid \exists a, b, c \in M(Rma \;\&\; Rba \;\&\; Rbc \;\&\; Rnc \;\&\; \mathfrak{M}, a \Vdash 1 \;\&\; \mathfrak{M}, b \Vdash 2 \;\&\; \mathfrak{M}, c \Vdash 3)\}$.
- $V^\blacktriangledown = V_{\restriction M^\blacktriangledown}$.

It follows by induction that for all $m \in M^\blacktriangledown$, for all uni-modal formulas ψ, $\mathfrak{M}, m \Vdash \psi^t$ iff $\mathfrak{M}^\blacktriangledown, m \Vdash \psi$. Only the step for $\Diamond$ is interesting. That $\mathfrak{M}, m \Vdash \psi^t$ implies $\mathfrak{M}^\blacktriangledown, m \Vdash \psi$ is straightforward. For the converse, suppose $\mathfrak{M}^\blacktriangledown, m \Vdash \Diamond\psi$. This means:

$$\exists n \in M^\blacktriangledown(R^\blacktriangledown mn \;\&\; \mathfrak{M}^\blacktriangledown, n \Vdash \psi).$$

As $n \in M$, we have that Rwn, hence by the definition of $R^\blacktriangledown$ we have that

$$\exists n, a, b, c \in M(Rma \;\&\; Rba \;\&\; Rbc \;\&\; Rnc \;\& Rwn \;\&\; \mathfrak{M}^\blacktriangledown, n \Vdash \psi),$$

and moreover a, b, c satisfy 1, 2, and 3 respectively. By the inductive hypothesis $\mathfrak{M}, n \Vdash \psi^t$. Hence, as w denotes i and Rwn, and as all points in $M^\blacktriangledown$ satisfy 0, we have $\mathfrak{M}, n \Vdash 0 \wedge Pi \wedge \psi^t$. It follows that $\mathfrak{M}, m \Vdash (\Diamond\psi)^t$.

From this equivalence and $\mathfrak{M}, w \Vdash \mathsf{G}(0 \to \varphi^t)$ it follows that $\mathfrak{M}^\blacktriangledown \Vdash \varphi$, and we have established the right to left direction of our claim. ◀

The theorem follows directly from the claim. ∎

Clearly P played a crucial role in this proof — so if we want a PSPACE or lower satisfiability result we will need to add further restrictions "to the past" to tame temporal hybridization. We will learn how to do this in the following section.

Now for the upper bound. We will establish this by extending known results for Propositional Dynamic Logic with nominals. It is known that satisfiability for PDL enriched with both nominals and E is solvable in EXPTIME (see [PT91]). Moreover, as De Giacomo observes (see [De 95]), his results on PDL-like description languages containing the $\mathcal{O}$ ("one-of") operator show that the satisfiability problem for PDL with converse enriched with nominals is solvable in EXPTIME. Now, on *connected* frames — assuming a finite repertoire of atomic programs — the somewhere modality is definable in converse PDL. But to establish the upper bounds we want, we need to know that we can have access to both converse programs *and* E on *arbitrary* frames and still stay in EXPTIME. And in fact, we can. Once again, we make use of a spypoint argument, but this time to obtain an upper bound:

THEOREM 3.5
The satisfiability problem for Propositional Dynamic Logic with converse enriched with both nominals and E is solvable in EXPTIME.

PROOF. Let ξ be a formula in this language. Without loss of generality, we may assume that converse is only applied to atomic programs. We will transform ξ into a formula without occurrences of E and then use de Giacomo's result. Let s be a nominal and σ be an atomic program not occurring in ξ. Define ξ^t by first recursively replacing every occurrence of $\mathsf{E}\varphi$ in ξ by $\langle\sigma^{-1}\rangle(s \wedge \langle\sigma\rangle\varphi)$, obtaining a formula without occurrences of E. Now transform the programs occurring inside the modalities in ξ by replacing atomic programs p (converse programs p^{-1}) by $p; \langle\sigma^{-1}\rangle s?$ (respectively $p^{-1}; \langle\sigma^{-1}\rangle s?$). We claim that ξ is satisfiable iff $\xi^t \wedge \langle\sigma^{-1}\rangle s$ is satisfiable. Since $\xi^t \wedge \langle\sigma^{-1}\rangle s$ is in the language covered by de Giacomo's EXPTIME-algorithm, this claim proves the theorem.

The left to right direction of the claim is obvious: just add a new state to the model, make s true there, and let that state be σ-connected to all other states.

For the other direction, let $\mathfrak{M} = \langle M, R_i, V\rangle_{i\in I}$ satisfy $\xi^t \wedge \langle\sigma^{-1}\rangle s$ at w and $V(s) = \{\mathbf{s}\}$. Let $\mathfrak{M}^-$ be the submodel of $\mathfrak{M}$ obtained by restricting the universe M to the set $M^- = \{x \in M \mid R_\sigma \mathbf{s} x\}$. We claim that for all subformulas ψ of ξ, for all $x \in M^-$,

$$\mathfrak{M}^-, x \Vdash \psi \iff \mathfrak{M}, x \Vdash \psi.$$

As $w \in M^-$, this would provide us with the desired result. The proof of the claim goes by the usual double induction needed for inductive proofs in PDL. The proof is straightforward given the following observations:

- the translation of E formulas works because we restricted the model to successors of the spypoint $\mathbf{s}$.
- $\mathfrak{M}^-$ is a *generated* submodel of $\mathfrak{M}$ for the new "atomic" programs $p; \langle\sigma^{-1}\rangle s?$ and $p^{-1}; \langle\sigma^{-1}\rangle s?$ occurring in ξ^t.

This finishes the proof of the theorem. ∎

COROLLARY 3.6
1. The satisfiability problem for nominal tense logic with E over arbitrary frames is solvable in EXPTIME.
2. The satisfiability problem for nominal tense logic with E over transitive frames is solvable in EXPTIME.

PROOF. For the arbitrary frame case, we define a linear translation $(.)^t$ of formulas of nominal tense logic (without @) into formulas of nominal PDL with converse as follows: $(a)^t{=}a$ for all atoms a, $(\neg\xi)^t = \neg(\xi)^t$, $(\xi \wedge \theta)^t = (\xi)^t \wedge (\theta)^t$, $(\mathsf{F}\xi)^t = \langle r\rangle(\xi)^t$, $(\mathsf{P}\xi)^t = \langle r^{-1}\rangle(\xi)^t$, and $(\mathsf{E}\xi)^t = \mathsf{E}(\xi)^t$. Here r is a fixed atomic program. It is easy to see that φ is satisfiable iff $(\varphi)^t$ is satisfiable.

For the transitive frame case, we define a linear translation $(.)^{\mathrm{tt}}$ which is identical to $(.)^t$ for atoms and booleans, but handles F and P as follows: $(\mathsf{F}\xi)^{tt} = \langle r; r^*\rangle(\xi)^{tt}$, and $(\mathsf{P}\xi)^{tt} = \langle r^{-1}; (r^{-1})^*\rangle(\xi)^{tt}$. It is easy to see that φ is satisfiable over a transitive frames iff $(\varphi)^{tt}$ is satisfiable. ∎

Hence, temporal hybridization on transitive frames still moves us outside the complexity class of the original logic. But from now on things get better. Two main kinds of transitive structures are usually considered as standard representations of time: strict total orders for linear time, and transitive trees for branching time. In the next two sections we will prove that in both cases hybridization is tamed. With this we mean to say that the expansion with nominals and @ does *not* lead to an increase in complexity.

3.2 Linear time

For many applications, time is modeled as linear. Of course, choosing linearity leaves many interesting options open, such as whether density or discreteness holds, and whether or not initial and final points in time exist. But for our purposes, such choices are irrelevant: the complexity of the satisfiability problem for nominal tense logic *over any subclass* of the class of strict total orders (that is, frames that are irreflexive, transitive, and trichotomous ($\forall s, t \in W(sRt \vee s = t \vee tRs)$) is the same as for Priorean tense logic. But given our remarks in Section 2 this should not come as a surprise: over strict total orders, F and P are strong enough to define D, as $\mathsf{D}\varphi \leftrightarrow (\mathsf{F}\varphi \vee \mathsf{P}\varphi)$ is valid. Thus we can eliminate all occurrences of nominals (and @ and E) by simulating them using D; in effect, we do the reverse of what we did in Theorem 2.1. Some care has to be taken to avoid a blow up in formula size during the elimination process, but it is easy to define an inductive replacement similar to that used in the proof of Theorem 2.1 that avoids such problems. Thus:

THEOREM 3.7
Let S be a subclass of the class of strict total orders. The complexity of the S-satisfiability problem is (up to a polynomial factor) the same for nominal tense logic (with or without @, E, or D) as it is for Priorean tense logic.

On many natural linear flows of time (for example, the class of all strict total orders, or the class containing only $\langle \mathbb{Q}, < \rangle$) the complexity of the satisfiability problem for the Priorean language is NP-complete [ON80, SC85], that is, no worse than propositional calculus, so nominal tense logic inherits these results.

Theorem 3.7 may seem a bit like cheating. We don't pay any computational cost, but this is because hybridization over strict total orders does not increase the expressive power at our disposal. This is true, but we think it misses the point. In many applications of temporal logic, reference to times is not an optional extra, it's fundamental. To model such problems naturally, we need formalisms which allow us to deal with temporal reference directly. Adding @ and nominals gives us what seems to be the correct level of abstraction needed. It is this level of abstraction (rather than the lower level offered by D) that needs to be isolated and explored. We'll return to this point in Section 5, when we discuss interval-based logic.

3.3 Branching time

In discussions of temporal logic in philosophy, natural language semantics, and computer science, the following intuition plays an important role: while several possible futures may be allowed, the past has a linear structure. To put it another way, it is often assumed that time has a tree-like structure, with the branching occurring only towards the future, never towards the past. Call a directed graph $\langle T, < \rangle$ a *tree* if it is acyclic and connected, and every node has at most one predecessor. A *transitive tree* is the transitive closure of a tree. In this section we are interested in the satisfiability problem for nominal tense logic over such frames.

Note that the spypoint argument used to prove EXPTIME-hardness in Theorem 3.4 *will not work* for transitive trees: the encoding of transitivity in the model $\mathfrak{M}^{4s}$ made crucial use of branching towards the past (points in M^1 look back to points in M^0 and M^2). And in fact, if we demand that our frames are transitive *trees*, we tame P and drop back into PSPACE. The key intuition comes from inspecting the structure of the past in a transitive tree: if we generate the submodel by the converse of the accessibility relation from any point in a transitive tree, we obtain a strict linear order. Thanks to this property it will be easy to satisfy past formulas in small structures. Future formulas will require more work, particularly when they interact with nominals. Nonetheless, we will be able to show that all the required computations can be carried out in PSPACE.

NP-*Complete Subfragments* Before we prove the PSPACE-result for nominal tense logic on transitive trees, let's look at possible further reductions in complexity on this class of frames. Consider the formula $\mathsf{F}i \wedge \mathsf{G}(\mathsf{F}i \rightarrow \varphi)$. It says that there is a state named i in the future and that φ holds at every state between now and the state named i. In other words, $\mathsf{F}i \wedge \mathsf{G}(\mathsf{F}i \rightarrow \varphi)$ says the same thing as $\mathsf{Until}(i, \varphi)$.

Consider the subfragment of nominal tense logic in which we only have future formulas of the form $\mathsf{Until}(i, \varphi)$, for one fixed nominal i (the rest of the language remains unchanged). For this fragment every satisfiable formula can be satisfied in a model of size polynomial in the length of the formula, yielding NP-completeness. This is shown by a simple submodel generation argument. Let $\mathfrak{M}, m \Vdash \varphi$. If $\mathfrak{M}, m \Vdash \mathsf{F}i$, create a submodel by considering the i-named state plus all its predecessors; otherwise create a submodel by taking m plus all its predecessors. In both cases we obtain a linear model which still satisfies φ at m. Since the models are linear, we can now use standard techniques from temporal logic to create a polysize model for φ. Here again, we obtained a reduction in complexity by *guarding* the modalities: we only allowed formulas of the form $\mathsf{G}(\mathsf{F}i \rightarrow \varphi)$.

THEOREM 3.8
Let $\mathcal{L}$ be the sublanguage of nominal tense logic in which every occurrence of a formula $\mathsf{F}\psi$ is of the form $\mathsf{F}(\mathsf{F}i \wedge \varphi)$, for some i, φ. Every $\mathcal{L}$ formula which is satisfiable on a transitive tree is satisfiable on a transitive tree of size polynomial in the length of the formula.

A simple inspection of the above argument shows that the restriction to just one nominal as guard is not needed to obtain the result. By distributing out disjunctions the same result holds for formulas of the form $\mathsf{F}(\mathsf{F}(i_1 \vee \ldots \vee i_k) \wedge \varphi)$. These formulas are used in the proof of the next theorem to distinguish future formulas that need their witnessing state before any named state from ones which don't, as follows. Suppose in $\mathfrak{M}$ there are only finitely many states named by a nominal, say $i_1, \ldots, i_k$. Let $\mathfrak{M}, m \Vdash \mathsf{F}\varphi$. Then either $\mathfrak{M}, m \Vdash \mathsf{F}(\mathsf{F}(i_1 \vee \ldots \vee i_k) \wedge \varphi)$ or $\mathfrak{M}, m \Vdash \mathsf{F}(\neg \mathsf{F}(i_1 \vee \ldots \vee i_k) \wedge \varphi)$. If the latter is false, this means that φ has to be true between m and some named state.

Theorem 3.9
The satisfiability problem for nominal tense logic over the class of transitive trees is PSPACE–complete.

Proof. PSPACE-hardness follows from results in [Lad77]: the satisfiability problem for ordinary uni-modal logic over transitive trees is already PSPACE-hard. The real work is to prove the PSPACE upper bound.

Our argument will be similar to the one used in Theorem 3.3, but now we should construct the appropriate tree structure in the game played by ∀belard and ∃loise. Instead of Hintikka sets, the players will use sequences of Hintikka sets which will take the role of partial branches in the model. But let us set up the game; first we define the following notions.

For any formula ξ in the nominal tense language, define the closure set $Cl(\xi)$ as the smallest set containing ξ, closed under subformulas, single negation and under the following rule: if $i \in Cl(\xi)$ then $\mathsf{P}(i) \in Cl(\xi)$.

Let ξ be fixed. A *thread* is a finite labelled frame $\langle T, <, l\rangle$ such that:

1. $<$ is a weak total order (i.e., $<$ is transitive and trichotomous),
2. $l : T \to Pow(Cl(\xi))$,
3. l labels with maximal consistent Hintikka sets over $Cl(\xi)$,
4. if $i \in l(x)$ then $\mathsf{P}i \notin l(x)$,
5. $|T| \leq |\{\mathsf{P}\varphi \in CL(\xi)\} \cup \{\mathsf{F}\varphi \in CL(\xi)\}| + 1$,
6. (Future coherence) if $\mathsf{F}\varphi \notin l(x)$ and $x < y$, then $\{\mathsf{F}\varphi, \varphi\} \cap l(y) = \emptyset$,
7. (Past saturation) $\mathsf{P}\varphi \in l(y)$ iff for some x, $x < y$ and $\varphi \in l(x)$.

The size $|t|$ of a thread $t = \langle T, <, l\rangle$ is $|T|$, and we say that $t_1 = \langle T_1, <_1, l_1\rangle$ and $t_2 = \langle T_2, <_2, l_2\rangle$ *fit at* $x \in T_1 \cap T_2$ iff $\langle T_1, <_1, l_1\rangle_{\restriction\{s \in T_1 \mid s \leq_1 x\}} = \langle T_2, <_2, l_2\rangle_{\restriction\{s \in T_2 \mid s \leq_2 x\}}$. Two threads *fit* if there exists an x such that they fit at x.

We first take care of the trivial case when $|\xi| = 1$. Then ξ is either an atom or a logical constant, and satisfiability is trivial. From now on we assume $|\xi| > 1$.

We can now specify the game for ∀belard and ∃loise. She should set up the playing board by specifying a collection M_0 of threads such that:

1. for some thread $t = \langle T, <, l\rangle \in M_0$, for some $x \in T$, $\xi \in l(x)$,
2. each two threads in M_0 fit,
3. there are no more threads in M_0 than the number of nominals in $Cl(\xi)$ plus one.
4. for each nominal i in $SF(\xi)$ there is a thread $t = \langle T, <, l\rangle \in M_0$, and $x \in T$ such that $i \in l(x)$,
5. let i be a nominal in $SF(\xi)$, t_1, t_2 two threads in M_0 and $x_1 \in T_1$, $x_2 \in T_2$ be such that $i \in l_1(x_1)$ and $i \in l_2(x_2)$, then $x_1 = x_2$ and t_1 and t_2 fit at x_1,

The rationale behind M_0 is that — when the threads are fitted together — it can be seen as a transitive tree model in which no nominal occurs in the label of two distinct states.

If $\exists$loise cannot set up the board she loses. Otherwise $\forall$belard chooses one of the threads in M_0, which will be the thread *in use* in the round. To start with, all the elements in the thread in use are *available*. In each round there will be a thread t in use, with a subset of its domain available for $\forall$belard to pick from. Furthermore we will keep a table S containing for each $\mathsf{F}\varphi$ formula in $Cl(\xi)$ a natural number $S(\mathsf{F}\varphi)$. At the start, for all $\mathsf{F}\varphi \in Cl(\xi)$, $S(\mathsf{F}\varphi) = |\xi|$. We are now ready to specify the movements of the players. In each round

- $\forall$belard points to an element x among the available elements in the thread t in use, and to a formula $\mathsf{F}\varphi$ in $l(x)$. x should be a maximal element containing $\mathsf{F}\varphi$. Formally, for all $y \in t$, $x < y$ implies $\mathsf{F}\varphi, \varphi \notin l(y)$ and furthermore, there is no thread $t'' = \langle T'', <'', l'' \rangle$ on the board, fitting t at x such that either $\mathsf{F}\varphi$ or φ are in $l''(y)$ for some $x <'' y$.
- The answer of $\exists$loise depends on the value recorded for the formula chosen by $\forall$belard.
  - ▶ If $S(\mathsf{F}\varphi) > 1$, then she should play a thread $t' = \langle T', <', l' \rangle$ such that
    - t' fits t at x,
    - there is $y \in T'$ such that $\varphi \in l'(y)$ and $x <' y$,
    - neither i nor $\mathsf{F}i$ is in $l'(z)$ for $x < z$, $i \in \mathsf{NOM}$,
    - $|\{z \in T' \mid x <' z\}| < S(\mathsf{F}\varphi)$.

    If she cannot present such a t', she loses. Otherwise, t' becomes the thread in use and $\{z \in T' \mid x <' z\}$ the available Hintikka sets for $\forall$belard. Furthermore $S(\mathsf{F}\varphi)$ is updated to $|\{z \in T' \mid x <' z\}|$.
  - ▶ If $S(\mathsf{F}\varphi) = 1$, then $\mathsf{F}\varphi$ has been played by $\forall$belard before. $\exists$loise should pick the Hintikka set e' containing φ that fixed the defect the last time $\forall$belard challenged $\mathsf{F}\varphi$. If $t' = \langle T_{\restriction\{z \in T \mid z \leq x\}} \cup \{n\}, < \cup\{(y, n) \mid y \in T\}, l \cup (n, e') \rangle$ for $n \notin T$ is a thread she wins. Otherwise she loses.

Notice that the values in S decrease as $\forall$belard repeats choices of $\mathsf{F}\varphi$ (which he cannot avoid as $Cl(\xi)$ is finite). This ensures termination of the game in at most $|\{\mathsf{F}\varphi \in Cl(\xi)\}| * |\xi|$ steps.

To prove the theorem we should now establish that $\exists$loise has a winning strategy in the game described above if and only if ξ is satisfiable; and then observe that in each game only a polynomial number of Hintikka sets are displayed on the board. (The extra information contained in the table S can obviously be coded using just linear space in the length of ξ.)

Claim: $\exists$loise has a winning strategy for the ξ-game iff ξ is satisfiable in a model based in a transitive tree frame.

Proof of Claim.
[$\Rightarrow$] Suppose $\exists$loise has a winning strategy in the ξ-game. We construct a model $\mathfrak{N} = \langle N, R, V \rangle$ as follows. Let M be the set of threads ever played by $\exists$loise in her winning strategy. $\mathfrak{N}$ is obtained by "superposing" one by one any two threads in M at their fitting point. More precisely, given two threads t and t' there is a maximal

point x such that t and t' fit. Hence for any thread t we can identify a maximal point x such that t fits with some other thread at x. The elements above x are "unique" to t; ensure uniqueness by renaming and let M' be the set of threads obtained in such a way. Notice that after such a renaming, for any element x of any thread in M' we can uniquely identify a Hintikka set corresponding to x, and for any two elements x, y in threads $t = \langle T, <, l\rangle$, $t' = \langle T', <', l'\rangle$ in M, if $x, y \in T \cap T'$ then $x < y$ iff $x <' y$.

Now let N be the union of all elements in threads in M', R be the union of the $<$ relations, and let V assign to an atomic symbol a the elements x in N such that a is in the Hintikka set corresponding to x. If i is a nominal not in $SF(\xi)$ then $V(i)$ is any singleton set in N. Since during the setting up of the board nominals in M_0 are assigned to unique elements and ∃loise never plays threads with new Hintikka sets containing nominals in her winning strategy, V is a hybrid valuation.

We claim that the truth lemma for elements in $Cl(\xi)$ holds for $\mathfrak{N}$, i.e., let $\varphi \in Cl(\xi)$, and s be any element in N, then $\mathfrak{N}, s \Vdash \varphi$ iff $\varphi \in H(s)$, where for any x in N, $H(x)$ is the Hintikka set corresponding to x.

The interesting case of the truth lemma are the modalities. By condition 7 in the definition of threads, elements in M' are past saturated, and the structure of threads has been preserved in $\mathfrak{N}$, hence the truth lemma for $\mathsf{P}\psi$ holds straightforwardly.

Suppose $\mathsf{F}\psi \in H(s)$. But $s \in t$ for some t in M (disregarding renaming), and ∀belard had a chance to chose each $\mathsf{F}\psi$ formula in any Hintikka set in M, at least for s which were maximal for formula $\mathsf{F}\psi$. If s is not maximal for $\mathsf{F}\psi$, then it has a maximal successor s' containing $\mathsf{F}\psi$, which could be challenged by ∀belard. As ∃loise has a winning strategy she should be able to answer the challenge and produce a thread t' fitting t at s'. By the condition of the game, t' will contain a Hintikka set s'' with ψ in its label set, such that $s' <' s''$. By the definition of $\mathfrak{N}$, transitivity, and IH, t' will contribute the needed witness into N.

For the converse we argue by contraposition. Suppose $\mathsf{F}\psi \notin H(s)$ and let s' be such that $R(s, s')$. But then there is a thread $t = \langle T, <, l\rangle \in M$ such that $s, s' \in T$ and $s < s'$. By condition 6, on threads ψ does not belong to $H(s')$. By IH, $\mathfrak{N}, s' \nVdash \varphi$. As s' was a generic successor of s we have $\mathfrak{N}, s \nVdash \mathsf{F}\varphi$.

The threads $t = \langle T, <, l\rangle$ used in the construction of $\mathfrak{N}$ might contain *clusters*, i.e., maximal (by inclusion) non empty sets S of T such that for all $x, y \in S$, $x < y$. But we can unravel them by a standard technique, preserving satisfaction. Notice that worlds labelled by nominals will never appear in clusters by items 4 and 7 in the definition of threads and hence they will not be duplicated by the unraveling. The unraveled model will be a transitive tree. This completes the left to right part of the claim.

[⇐] We should now prove that if ξ is satisfiable, then ∃loise has a winning strategy. Let $\mathfrak{M}$ be a transitive tree model such that $\mathfrak{M}, w \Vdash \xi$. And let $\mathfrak{M}^f = \langle M^f, R^f, V^f\rangle$ be a transitive filtration of $\mathfrak{M}$ under $Cl(\xi)$. By properties of filtrations we know that $\mathfrak{M}^f$ is finite and R^f is transitive. Furthermore R^f has maximal (and minimal) elements, i.e., those $|m| \in M^f$ such that for all $|m'| \in M^f$, $R^f|m||m'| \Rightarrow R^f|m'||m|$. Also, elements of M^f can be uniquely identified with Hintikka sets over $Cl(\xi)$ (and we will treat them as such). But $|M^f|$ can still be exponential in $|\xi|$.

For each element $|m| \in M^f$ we can built a thread t as follows. For each $\mathsf{P}\varphi$ in $|m|$ choose a minimal predecessor $|m'|$ of $|m|$ which satisfies φ. Order t by R^f. Since the filtration comes from a transitive tree, a standard argument shows that t is past saturated and hence a thread. Furthermore $|t|$ is linear in $|\xi|$. We call such t a thread built from $|m|$ in M^f. Let φ be a formula satisfiable in $\mathfrak{M}^f$. Then a thread for φ is a thread t from some $|m|$, such that $\mathfrak{M}^f, |m| \Vdash \varphi$, but we require in addition that $|m|$ is R^f-maximal satisfying φ, and that t is *long*, i.e., for all other threads t' built from $|m'|$ such that $\mathfrak{M}^f, |m'| \Vdash \varphi$, $|t'| < |t|$.

Let t be a thread built from $|m|$ and let $|m'|$ be an element in t. Let $|m''|$ be an R^f-successor of m'. It is not difficult to prove that there is a thread t' built from $|m''|$ that fits t at m'. Furthermore t' is linear in $|\xi|$. From this fact,

(*) if t is a thread built from an element in $\mathfrak{M}^f$ and $|m|$ is any element in t such that $\mathfrak{M}^f, |m| \Vdash \mathsf{F}\varphi$. Then there is a thread for φ that fits t at $|m|$.

All this machinery is used to specify ∃loise's answers to challenges made by ∀belard. We first specify how ∃loise can set up the board. Clearly she can find past saturated threads containing ξ and all the nominals and clearly they also fit. But she has to do something extra. Since later in the game she cannot use threads containing nominals in new points she sometimes has to insert extra witnessing points in these threads. Suppose she wants to include a thread containing $|m|$, and suppose $\mathfrak{M}^f, |m| \Vdash \mathsf{F}\varphi$ but *all* $|m'|$ such that $R^f|m||m'|$ and $\varphi \in m'$ lie before a nominal occuring in ξ. Then she should include a witnessing $|m'|$ already in step 0. Theorem 3.8 and the remarks just following it state precisely that she can include all these witnesses and still use only a polynomial number of states (the exact required bound is easily computed from the proof). With this setup it is guaranteed that in the case the antecedent of $(*)$ above obtains, the needed fitting thread can be chosen not to contain any nominal or $\mathsf{F}i$ formula.

With ∃loise's groundwork taken care of, we now specify how ∃loise answers the challenges of ∀belard. Suppose that ∀belard points to a defect formula $\mathsf{F}\varphi$ in an element $|m|$ in t. If $S(\mathsf{F}\varphi) > 1$, she answers with a thread t' provided by (*), again choosing t' to be maximal and long. As argued above she can answer with a thread in which neither i nor $\mathsf{F}i$ appears in t' above the fitting point.

By obtaining threads from $\mathfrak{M}^f$, ∃loise can always answer the moves of ∀belard. What remains to be checked is the condition on decreasing lengths when ∀belard plays a repeated formula. So assume that ∀belard has in a previous round chosen $\mathsf{F}\varphi \in |m| \in t_1$, and that ∃loise answered with a long maximal thread t_2. Furthermore he is now choosing $\mathsf{F}\varphi \in |m'| \in t_3$. We know that ∃loise can produce a long maximal thread t_4 to answer the challenge. But it is immediate that if the size of t_4 above $|m'|$ is greater than the size of t_2 above $|m|$ then t_2 was not a long maximal thread.

Suppose now that $S(\mathsf{F}\varphi) = 1$ and let t_1 be the thread previously played by ∃loise as an answer, with $\varphi \in |m_1|$. Again by (*) we could produce a thread t' fitting t at m and containing φ in some element $|m'|$. Furthermore, because of the decreasing condition, we know that this time she can choose t' to be very short. Actually $|m|$ is at "one step" from the R^f-maximal cluster satisfying φ, and ∃loise can choose any element of such a cluster, in particular $|m_1|$. ◀

As we already mentioned, the game stops in time quadratic in the size of ξ, furthermore all the information ever played in the board can be encoded in polynomial space. Hence the theorem follows. ∎

What happens if we add E? Nothing — for we already have it: over transitive trees $\mathsf{E}\varphi$ can be defined to be $\varphi \vee \mathsf{F}\varphi \vee \mathsf{P}\varphi \vee \mathsf{PF}\varphi$.

Corollary 3.10
The transitive tree satisfiability problem for nominal tense logic with E is PSPACE-complete.

With this we close our analysis of the complexity of nominal tense logic. In the next two sections we discuss two other hybrid temporal logics for which we have obtained some preliminary results.

4 Nominal Until/Since logic

We first turn to two other operators often used in temporal logic: Until and Since. On any (Kripke or hybrid) model $\mathfrak{M} = \langle M, R, V\rangle$, they are defined as follows:

$$\mathfrak{M}, s \Vdash \mathsf{Until}(\varphi, \psi) \text{ iff } \exists t \in M(sRt \;\&\; \mathfrak{M}, t \Vdash \varphi \;\&\; \forall r \in M(sRrRt \Rightarrow \mathfrak{M}, r \Vdash \psi))$$
$$\mathfrak{M}, s \Vdash \mathsf{Since}(\varphi, \psi) \text{ iff } \exists t \in M(tRs \;\&\; \mathfrak{M}, t \Vdash \varphi \;\&\; \forall r \in M(tRrRs \Rightarrow \mathfrak{M}, r \Vdash \psi)).$$

Using Until we can define $\mathsf{F}\varphi := \mathsf{Until}(\varphi, \top)$ and similarly, using Since we can define $\mathsf{P}\varphi := \mathsf{Since}(\varphi, \top)$. It follows by our earlier results on nominal tense logic that enriching a language containing both Until and Since with even a single nominal results in an EXPTIME-hard satisfiability problem over both arbitrary and transitive frames. But it turns out that in this case, the introduction of a past operator is not to blame: already the pure future fragment is EXPTIME-hard. In what follows we will work with languages whose only primitive modality is Until, and treat F (and G) as defined symbols.

Now, it is well known that Priorean tense logic is not strong enough to define Until, even over linear frames, and the addition of nominals and @ does not help. On the other hand, unlike the difference operator, Until is not strong enough to simulate nominals (indeed, neither can Until and Since working together) on arbitrary structures. Thus enriching an Until-based language with nominals results in a genuine increase in expressive power; for example, many new classes of frames become definable. So: what is the complexity of nominal Until logic?

Theorem 4.1
The satisfiability problem over arbitrary frames for a Until logic extended with a single nominal (and no @) is EXPTIME-hard.

Proof. The proof strategy is the same as that used for Theorem 3.4: we shall reduce the EXPTIME-complete *global* satisfiability problem for ordinary uni-modal languages to the (local) satisfiability problem for a language containing the Until operator and at least one nominal (and no @) by means of a spypoint argument. First we define:

$$i\text{-}spy \;:=\; i \wedge \neg\mathsf{F}i \wedge \mathsf{FF}i \wedge \mathsf{G}\neg\mathsf{Until}(\mathsf{F}i, \neg i).$$

We shall use the following three observations in our proof. If i-spy is satisfied in a model $\langle M, R, V\rangle$ at a point s then:

1. $\neg Rss$.
2. $\exists m \in M(m \neq s \;\&\; Rsm \;\&\; Rms)$.
3. $\forall m, m' \in M(Rsm \;\&\; Rmm' \;\&\; Rm's \;\Rightarrow\; Rsm')$.

Observations 1 and 2 are obvious consequences of the first three conjuncts of *i-spy*. Observation 3 is the crucial one and requires comment. Any formula of the form $\neg\mathsf{Until}(\xi,\theta)$ is true at a point m in a model iff for every successor m' of m at which ξ is true, there is a point u between m and m' at which θ is *false*. So if $\neg\mathsf{Until}(\mathsf{F}i,\neg i)$ is true at m, this means that at every successor m' of m which precedes s (for that is what $\mathsf{F}i$ asserts) there is a point u between m and m' at which $\neg i$ is false. But there is only one point in the entire model where $\neg i$ is false, namely s. In short $\neg\mathsf{Until}(\mathsf{F}i,\neg i)$ guarantees that every successor m' of m which precedes s is also preceded by s. Prefixing G ensures that all successors m of s will have this property.

Next, we define the following translation function $(\cdot)^t$ from ordinary uni-modal formulas to formulas in a language containing the Until operator and at least one nominal i:

$$\begin{array}{rcl} p^t & = & p \\ (\neg\xi)^t & = & \neg\xi^t \\ (\xi\wedge\theta)^t & = & \xi^t\wedge\theta^t \\ (\Diamond\xi)^t & = & \mathsf{F}(\xi^t\wedge\mathsf{F}i). \end{array}$$

Clearly $(\cdot)^t$ is a linear reduction.

CLAIM: For any ordinary uni-modal formula φ, φ is globally satisfiable in a Kripke model iff *i-spy* $\wedge\,\mathsf{G}(\varphi^t)$ is satisfiable in a hybrid model.

PROOF OF CLAIM.
[$\Leftarrow$]. Suppose that $\mathfrak{M},s\Vdash$ *i-spy* $\wedge\,\mathsf{G}(\varphi^t)$, where $\mathfrak{M}=\langle M,R,V\rangle$ is a hybrid model. Clearly s must be the unique element in $V(i)$, and our observations concerning *i-spy* hold. Define $\mathfrak{M}^\blacktriangledown$ as follows

- $M^\blacktriangledown=\{m\in M\mid Rsm\ \&\ Rms\}$.
- $R^\blacktriangledown=R_{\restriction M^\blacktriangledown}$.
- $V^\blacktriangledown=V_{\restriction M^\blacktriangledown}$.

Note that by observation 2, $M^\blacktriangledown\neq\emptyset$. Next, note that for all $m\in M^\blacktriangledown$, for all ordinary uni-modal formulas ψ, $\mathfrak{M},m\Vdash\psi^t$ iff $\mathfrak{M}^\blacktriangledown,m\Vdash\psi$. This follows by induction on the structure of φ. The interesting case is the step for $\Diamond$:

$$\begin{array}{rl} & \mathfrak{M},m\Vdash\mathsf{F}(\psi^t\wedge\mathsf{F}i) \\ \Longleftrightarrow & \exists m'\in M(Rmm'\ \&\ \mathfrak{M},m'\Vdash\psi^t\ \&\ m'\neq s\ \&\ Rm's) \\ \Longleftrightarrow & \exists m'\in M^\blacktriangledown(Rmm'\ \&\ \mathfrak{M},m'\Vdash\psi^t\ \&\ m'\neq s\ \&\ Rm's) \\ \Longleftrightarrow & \exists m'\in M^\blacktriangledown(R^\blacktriangledown mm'\ \&\ \mathfrak{M}^\blacktriangledown,m'\Vdash\psi) \\ \Longleftrightarrow & \mathfrak{M}^\blacktriangledown,m\Vdash\Diamond\psi. \end{array}$$

(The key step in the argument is showing that line 2 implies line 3; that is, that $m'\in M^\blacktriangledown$. This is where we appeal to observation 3.) Now, by assumption $\mathfrak{M},s\Vdash$ *i-spy* $\wedge\,\mathsf{G}(\varphi^t)$. So for all $m\in M$ such that Rsm, we have $\mathfrak{M},m\Vdash\varphi^t$. That is, for all $m\in M^\blacktriangledown$, $\mathfrak{M},m\Vdash\varphi^t$. Hence by the equivalence just proved, for all $m\in M^\blacktriangledown$, $\mathfrak{M}^\blacktriangledown,m\Vdash\varphi$. In short, $\mathfrak{M}^\blacktriangledown\Vdash\varphi$, and we have proved the right to left direction of our claim.

[$\Rightarrow$]. Let $\mathfrak{M}\Vdash\varphi$, where $\mathfrak{M}=\langle M,R,V\rangle$ is an ordinary Kripke model. Let s be an element not belonging to M. We define a hybrid model $\mathfrak{M}^s$ as follows:

- $M^s = M \cup \{s\}$.
- $R^s = R \cup \{(s, m), (m, s) \mid m \in M\}$.
- $V^s = V \cup \{(i, \{s\})\}$.

(If there are nominals in the language besides i, we let V^s assign them arbitrary singletons sets; any such nominals are irrelevant to what follows.) We claim that for all $m \in M$, for all ψ, we have $\mathfrak{M}, m \Vdash \psi$ iff $\mathfrak{M}^s, m \Vdash \psi^t$. This follows by a simple induction. The only interesting step is for $\Diamond$:

$$\begin{array}{ll} & \mathfrak{M}, m \Vdash \Diamond\psi \\ \iff & \exists m' \in M(Rmm' \;\&\; \mathfrak{M}, m' \Vdash \psi) \\ \iff & \exists m' \in M^s(R^s mm' \;\&\; \mathfrak{M}^s, m' \Vdash \psi^t \;\&\; m' \neq s \;\&\; R^s m' s) \\ \iff & \mathfrak{M}^s, m \Vdash \mathsf{F}(\psi^t \wedge \mathsf{F}i) \\ \iff & \mathfrak{M}^s, m \Vdash (\Diamond\psi)^t. \end{array}$$

(To see that the third line implies the second, note that if $m' \neq s$ and $R^s m' s$, then $m' \in M$.) Now by assumption, $\mathfrak{M} \Vdash \varphi$. Hence by the equivalence just proved, for all $m \in M$, $\mathfrak{M}^s, m \Vdash \varphi^t$. Hence $\mathfrak{M}^s, s \Vdash \mathsf{G}(\varphi^t)$. We also need to show that $\mathfrak{M}^s, s \Vdash i\text{-}spy$. That the first three conjuncts of i-*spy* hold at s is clear. $\mathsf{G}\neg\mathsf{Until}(\mathsf{F}i, \neg i)$, its last conjunction, holds because s precedes every point in the model except itself. ◀

The theorem follows directly from the claim. ∎

Once again, however, if we give a temporal interpretation to Until we are more interested in satisfiability over transitive frames. Moreover, on transitive frames Until captures a much stronger sense of "betweenness" than it does over non-transitive frames. So: what is the complexity of nominal until logic over transitive frames? We don't have an answer to this. We cannot directly use the zig-zag between past and future used in Theorem 3.4 (we don't have a backward looking operator) and we don't see any other way of polynomially encoding transitivity.

However we do have a related result. Suppose we replace Until by the operator Until^+ with the following satisfaction definition. Let $\mathfrak{M} = \langle M, R, V \rangle$ be any (hybrid or Kripke) model and let R^+ denote the transitive closure of R, then

$$\mathfrak{M}, x \Vdash \mathsf{Until}^+(\varphi, \psi) \text{ iff } \exists y \in M(xRy \;\&\; \mathfrak{M}, y \Vdash \varphi \;\&\; \forall z \in M(xR^+zR^+y \Rightarrow \mathfrak{M}, z \Vdash \psi)).$$

That is, $\mathsf{Until}^+(\varphi, \psi)$ holds at x if φ holds at some x-successor y and for all z which are in between x and y, ψ holds, where "in between" now is not restricted to immediate successors and predecessors. In other words Until^+ captures the same strong sense of "betweenness" on arbitrary frames that Until captures on transitive frames. Note that the satisfaction definition of Until^+ is not first-order. So: what is the complexity of this variant of nominal until logic? As before we have:

THEOREM 4.2
The satisfiability problem over arbitrary frames for a language containing Until^+ and at least one nominal (but no @) is EXPTIME-hard.

PROOF. We shall reduce the same EXPTIME-complete problem, using the same translation function $(.)^t$, of Theorem 4.1 — but instead of a spypoint argument, we shall use what might be called a *sinkpoint* argument.

CLAIM: For any ordinary uni-modal formula φ, φ is globally satisfiable in a Kripke model iff $\varphi^t \wedge \mathsf{Until}^+(i, \varphi^t) \wedge \mathsf{F}(i \wedge \neg\mathsf{F}i)$ is satisfiable in a hybrid model.

PROOF OF CLAIM.
[$\Rightarrow$]. Let $\mathfrak{M} = \langle M, R, V\rangle \Vdash \varphi$. Without loss of generality we can assume that $\mathfrak{M}$ has a root w (simply take a submodel generated by any point w). Define $\mathfrak{M}^s$ as follows:

- $M^s = M \cup \{s\}$.
- $R^s = R \cup \{(x, s) \mid x \in M\}$.
- $V^s = V \cup \{(i, \{s\})\}$.

Note that s is a sinkpoint: instead of using s to spy on all the points in the model, we squeeze all the points we need between w and s. It is easy to show that $\mathfrak{M}^s, w \Vdash \varphi^t \wedge \mathsf{Until}^+(i, \varphi^t) \wedge \mathsf{F}(i \wedge \neg\mathsf{F}i)$.

[$\Leftarrow$]. Let $\mathfrak{M} = \langle M, R, V\rangle, w \Vdash \varphi^t \wedge \mathsf{Until}^+(i, \varphi^t) \wedge \mathsf{F}(i \wedge \neg\mathsf{F}i)$. Let $V(i) = \{s\}$. Define $\mathfrak{M}^\blacktriangledown$ as

- $M^\blacktriangledown = \{x \in M \mid x = w \text{ or } wR^+xRs\}$.
- $R^\blacktriangledown = R_{\restriction M^\blacktriangledown}$.
- $V^\blacktriangledown = V_{\restriction M^\blacktriangledown}$.

Note that $w \in M^\blacktriangledown$. Obviously, for all $x \in M^\blacktriangledown$, $\mathfrak{M}^\blacktriangledown, x \Vdash \varphi$ if and only if $\mathfrak{M}, x \Vdash \varphi^t$. Thus $\mathfrak{M}^\blacktriangledown \Vdash \varphi$, because for all $x \in M^\blacktriangledown$, $\mathfrak{M}, x \Vdash \varphi^t$. ◀

The theorem follows directly from the claim. ∎

For both, Theorem 4.1 and Theorem 4.2 matching EXPTIME upper bounds can be obtained by embedding into the loosely μ-guarded fragment [GW99], but we omit details.

Using ideas from Section 3.3, we can point to a subset of the language with an NP-complete satisfaction problem. Consider the formulas of the $\mathsf{Until}/\mathsf{Since}$ language equivalent to formulas of the form $\mathsf{Since}(i, \psi)$ or $\mathsf{Until}(i, \psi)$. It is immediate that, in any structure, these formulas are equivalent to $\mathsf{P}i \wedge \mathsf{H}(\mathsf{P}i \rightarrow \psi)$ and $\mathsf{F}i \wedge \mathsf{G}(\mathsf{F}i \rightarrow \psi)$, respectively. We can use then Theorem 3.7 to prove that the complexity on linear frames drops from PSPACE-complete for the full language (see [SC85]) to NP-complete for the fragment. By Theorem 3.8 a similar drop in complexity occurs on the class of transitive trees.

5 Referential interval logic

We will now examine what happens when we add nominals and @ to the interval-based logic of Halpern and Shoham (see [HS86]). This logic arises by abstracting from James Allen's system of temporal knowledge representation (see [All84]). As with our discussion of nominal tense logic over strict total orders, the complexity result we shall prove hinges on the observation that Halpern and Shoham's language is powerful enough to define the difference operator D. However we are going to work with a more interesting type of hybrid language, namely one containing two distinct sorts of nominals. That is, we will be working in a three-sorted modal language.

The language of Halpern and Shoham builds formulas from a collection of atomic symbols ATOM using boolean operators and the primitive modalities $\langle A\rangle$, $\langle\overline{A}\rangle$, $\langle B\rangle$, $\langle\overline{B}\rangle$, $\langle E\rangle$ and $\langle\overline{E}\rangle$. As it is a traditional (unsorted) modal language, ATOM contains only the symbols in PROP, that is, propositional variables p, q, r, and so on. The language is interpreted on interval structures built over strict total orders. Let $\langle T,<\rangle$ be a strict total order and let $I(\langle T,<\rangle)$ be the set of all closed intervals $[t,t'] = \{s \in T : t \leq s \leq t'\}$ on $\langle T,<\rangle$. A Kripke model $\mathfrak{M}$ is a triple $\langle T,<,V\rangle$. Here V is a valuation, that is, a mapping $V : \mathsf{ATOM} \to Pow(I(\langle T,<\rangle))$. The satisfaction definition is:

$$\begin{array}{lll}
\mathfrak{M},[t_1,t_2] \Vdash a & \text{iff} & [t_1,t_2] \in V(a),\ \text{for all atoms } a\\
\mathfrak{M},[t_1,t_2] \Vdash \neg\varphi & \text{iff} & \mathfrak{M},[t_1,t_2] \nVdash \varphi\\
\mathfrak{M},[t_1,t_2] \Vdash \varphi\wedge\psi & \text{iff} & \mathfrak{M},[t_1,t_2] \Vdash \varphi \text{ and } \mathfrak{M},[t_1,t_2] \Vdash \psi\\
\mathfrak{M},[t_1,t_2] \Vdash \langle A\rangle\varphi & \text{iff} & \exists t_3 : t_2 < t_3 \text{ and } \mathfrak{M},[t_2,t_3] \Vdash \varphi\\
\mathfrak{M},[t_1,t_2] \Vdash \langle\overline{A}\rangle\varphi & \text{iff} & \exists t_3 : t_3 < t_1 \text{ and } \mathfrak{M},[t_3,t_1] \Vdash \varphi\\
\mathfrak{M},[t_1,t_2] \Vdash \langle B\rangle\varphi & \text{iff} & \exists t_3 : t_3 < t_2 \text{ and } t_1 \leq t_3 \text{ and } \mathfrak{M},[t_1,t_3] \Vdash \varphi\\
\mathfrak{M},[t_1,t_2] \Vdash \langle\overline{B}\rangle\varphi & \text{iff} & \exists t_3 : t_2 < t_3 \text{ and } \mathfrak{M},[t_1,t_3] \Vdash \varphi\\
\mathfrak{M},[t_1,t_2] \Vdash \langle E\rangle\varphi & \text{iff} & \exists t_3 : t_1 < t_3 \text{ and } t_3 \leq t_2 \text{ and } \mathfrak{M},[t_3,t_2] \Vdash \varphi\\
\mathfrak{M},[t_1,t_2] \Vdash \langle\overline{E}\rangle\varphi & \text{iff} & \exists t_3 : t_3 < t_1 \text{ and } \mathfrak{M},[t_3,t_2] \Vdash \varphi.
\end{array}$$

Let's now sort this language to make it referential. We could introduce a "generic" nominal naming arbitrary intervals — but to make things more interesting, let's take a more fine grained approach. We will add two different sorts of nominals: *stretched interval nominals*, and *point interval nominals*. Stretched interval nominals will be used to name extended time periods, and point interval nominals will denote point-like intervals. So, let SNOM and PNOM be denumerably infinite sets that are mutually disjoint, and disjoint from PROP. We represent the elements of SNOM as e, d, c, and those of PNOM by i, j, k, and we redefine ATOM to be $\mathsf{PROP} \cup \mathsf{SNOM} \cup \mathsf{PNOM}$. The only other change we will make is to add @; either type of nominal can appear as a subscript.

Let $S(\langle T,<\rangle) = \{[t,t'] \in I(\langle T,<\rangle) \mid t \neq t'\}$ be the set of stretched intervals. Let $P(\langle T,<\rangle) = \{[t,t'] \in I(\langle T,<\rangle) \mid t = t'\}$ be the set of point intervals. We now define a *hybrid valuation* to be a map $V : \mathsf{ATOM} \to Pow(I(\langle T,<\rangle))$ assigning single subsets of $S(\langle T,<\rangle)$ to elements in SNOM and single subsets of $P(\langle T,<\rangle)$ to elements in PNOM. As usual, @ means "jump to the interval denoting the subscript and test for the truth of its argument there." This three-sorted language can be thought of as the propositional fragment of James Allen's system of temporal knowledge representation [All84]. In particular, what Allen would write as $\mathsf{Holds}(e,\varphi)$ corresponds to $@_e\varphi$ in the hybrid language.

We will make some further remarks on the connections with Allen's work shortly, but let's now look at the complexity of this system. In fact, sorting and adding @ has not made the satisfiability problem any worse. As we already said, the key point is that D is definable:

$$\mathsf{D}\varphi \ := \bigvee_{\pi\in\{A,\overline{A},B,\overline{B},E,\overline{E},L,\overline{L},D,\overline{D},O,\overline{O}\}} \langle\pi\rangle\varphi.$$

This definition makes use of the following defined operators:

$$\langle L\rangle\varphi := \langle A\rangle\langle A\rangle\varphi, \quad \langle D\rangle\varphi := \langle B\rangle\langle E\rangle\varphi, \quad \langle O\rangle\varphi := \langle E\rangle\langle\overline{B}\rangle\varphi,$$
$$\langle\overline{L}\rangle\varphi := \langle\overline{A}\rangle\langle\overline{A}\rangle\varphi, \quad \langle\overline{D}\rangle\varphi := \langle\overline{B}\rangle\langle\overline{E}\rangle\varphi, \quad \langle\overline{O}\rangle\varphi := \langle B\rangle\langle\overline{E}\rangle\varphi.$$

If you spell out what each of the disjunctions in the definition of D say, you will see that between them they cover 12 of the 13 possible relations between intervals in an interval structure $I(\langle T, <\rangle)$. The missing relation is equality, and if we add a disjunction for this we obtain E. That is, $\mathsf{E}\varphi := \varphi \vee D\varphi$.

Now, if we had simply introduced a generic interval nominal, we could use D to simulate nominals and @ exactly as we did in our discussion of nominal tense logic over linear frames. But we have to draw a distinction between stretched and point nominals. We do so as follows. Observe that $\langle B\rangle\top$ is true on precisely the elements of $S(\langle T, <\rangle)$, while $\langle B\rangle\top$ is false on precisely the elements of $P(\langle T, <\rangle)$. So the formula $\mathsf{E}(p \wedge \langle B\rangle\top) \wedge \mathsf{A}(p \rightarrow \neg\mathsf{D}p)$ forces the propositional variable p to act like a stretched interval nominal, and $\mathsf{E}(p \wedge \neg\langle B\rangle\top) \wedge \mathsf{A}(p \rightarrow \neg\mathsf{D}p)$ forces it to act like a point interval nominal. Then, as we discussed in Section 3.2, by substituting ordinary propositional symbols for nominals, and using these definitions to make them behave appropriately, we can show that:

THEOREM 5.1
Let S be a subclass of the class of strict total orders. The complexity of the satisfiability problem over the interval structures built over the frames in S is the same for the three-sorted interval-based logic (with @) as it is for the logic of Halpern and Shoham.

But there is a difference between this result and our earlier result for nominal tense logic over linear frames: the satisfiability problem for Halpern and Shoham's logic is typically much harder. For example, as Halpern and Shoham show, validity is not recursively enumerable if we build our intervals over $\langle\mathbb{R}, <\rangle$, the real numbers in their usual order.

In our view, this suggests two main paths for further work. The first is to look at the complexity of weaker hybrid interval logics in which nominals and @ are added as *primitives*. The most striking feature of Allen's work is his use of Holds (without Holds, Allen's system is a fairly orthodox first-order theory of intervals). Moreover, the problem of solving constraints on temporal reference is one of the key tasks in temporal knowledge representation. But the system of Halpern and Shoham isn't built on the idea of reference — temporal reference is a side-effect of the definability of D. And this means that once we start dropping modalities in the hope of finding better behaved subfragments, we lose the ability to refer. It seems to us that it is much more in the spirit of Allen's work to add point interval nominals, stretched interval nominals, and @, right from the start, and then to see what can be achieved using a more restricted collection of operators (for example, the four interval operators discussed in [Ben83]).

A second approach is to *increase* the expressivity still further. In particular, it seems interesting to go to a richer hybrid language which allows explicit quantification over nominals. If this is done, the resulting system perspicuously captures the full version of Allen's system (Allen allows explicit quantification over interval terms). Hybrid languages that allow explicit quantification over nominals (and indeed, other sorts) have long been studied; see in particular [Bul70] and [PT91]. Given that the logic of Halpern and Shoham is undecidable (or even highly undecidable) for many important classes of interval structure, we have nothing to lose on complexity theoretic grounds by adding explicit quantification over nominals. And we do have something to gain, namely expressivity: as is shown in [Ven90], the logic of Halpern and Shoham is not expressively complete over the class of dense linear orders, and in fact, neither is any interval logic based on a finite collection of interval tense operators.

6 Concluding remarks

In this paper we investigated the complexity of the satisfiability problem for a number of hybrid temporal logics. Our main results dealt with nominal tense logic, that is, Priorean tense logic enriched with nominals and possibly @, E, and D as well. The following table summarizes our results here:

Class of frames	[F, P]	[F, P, @, Nom]	[F, P, E], [F, P, E, @]	[F, P, E, Nom], [F, P, D]
All frames	PSPACE	EXPTIME	EXPTIME	EXPTIME
Transitive	PSPACE	EXPTIME	EXPTIME	EXPTIME
Strict Total Orders	NP	NP	NP	NP
Transitive trees	PSPACE	PSPACE	PSPACE	PSPACE

Some comments. Hybrid temporal logic has an EXPTIME-hard satisfiability problem over both arbitrary frames (Theorem 3.2) and transitive frames (Theorem 3.4). These results hold even without @, and only one nominal is needed to establish them. Matching EXPTIME upper bounds hold for nominal tense logic, over both arbitrary and transitive frames, even if E is added (Corollary 3.6). Over any class of strict total orders, the satisfiability problem for nominal tense logic is the same as that of Priorean tense logic (Theorem 3.7); for example, for the class of all strict total orders, nominal tense logic with E has an NP-complete satisfiability problem. Over trees, nominal tense logic (both with and without @) is PSPACE-complete (Theorem 3.9), even when extended with E or D (Corollary 3.10).

With respect to pure-future fragments, Theorem 3.1 and Theorem 3.3 show that hybridization does not damage the fundamental complexity results for modal logics.

We turned then to languages more expressive than tense logic. We obtained two EXPTIME characterizations for Until logics and briefly discussed a hybrid version of Halpern and Shoham interval logics. One general comment. It is striking how little is known about complexity results for these languages. Results from computer science for language containing Until typically deal with quite restricted (usually tree-like) classes of frames. As for interval logic, [Ben83] discusses a wide range of possible constraints on interval structure, but it seems that few of these have been investigated from a complexity theoretic perspective. Given the high expressive power of these logics, many decidable logics in this family will probably turn to be EXPTIME; still, a finer analysis of the trade off between complexity and expressive power in this area would be extremely useful.

To conclude the paper we note some consequences of our results for description logic. We have already mentioned in Section 1 that the description language $\mathcal{ALC}$ (without axioms) is a notational variant of multi-modal logic. Assertional axioms over $\mathcal{ALC}$ (that is, the formulas listed in the A-Box) can be formulated in a restricted fragment of multi-modal logic enriched with nominals and @, namely the fragment in which nominals cannot be freely used in formulas, but can only occur as indices on the @ operator. The description logic $\mathcal{ALCO}$ moves closer towards full multi-modal logic enriched with nominals and @. $\mathcal{ALCO}$ allows the formation of concepts using the $\mathcal{O}$ ("one-of") operator: an expression of the form $\mathcal{O}(i, \ldots, k)$ corresponds to a disjunction of nominals $i \vee \cdots \vee k$, and an expression of the form $\mathcal{O}(i)$ to the nominal i (see [BT98]). Thus the full language of multi-modal logic enriched with nominals and @ is essentially a version of $\mathcal{ALCO}$ in which no restrictions are made on the use of assertional statements.

Now, as was observed in [ABM99], Theorem 3.1 extends straightforwardly to the multi-modal case. Viewed from a description logic perspective, this tells us that dropping all restrictions on the use of A-box descriptions does not move us out of PSPACE (for the concept satisfiability problem). That is, we can work in a "one-level" language (let's call it $\mathcal{ALCO}$@) in which A-box descriptions can be freely combined with the usual description logic machinery and $\mathcal{O}$, and remain in PSPACE.[3]

Compare this with what happens if we add converse roles. If we start with a language of $\mathcal{ALCO}$ that contains only one role symbol R we have a PSPACE-complete satisfiability problem, and this complexity is not modified no matter how many further ordinary role symbols we add. But Theorem 3.2 tells us that if we add the *converse* role symbol R^{-1} we immediately jump to EXPTIME.

One final remark: the spypoint argument used throughout the paper seems of interest in its own right. We have seen that the technique is useful in hybrid logic for establishing both upper and lower bounds, and we think it may be useful in description (and feature) logic too.

References

[ABM99] C. Areces, P. Blackburn, and M. Marx. A road-map on complexity for hybrid logics. In J. Flum and M. Rodríguez-Artalejo, editors, *Computer Science Logic*, number 1683 in LNCS, pages 307–321. Springer, 1999.

[AdR99] C. Areces and M. de Rijke. Accounting for assertional information. Manuscript, 1999.

[All84] J. Allen. Towards a general theory of action and time. *Artificial Intelligence*, 23(2):123–154, July 1984.

[AvBN98] H. Andréka, J. van Benthem, and I. Németi. Modal languages and bounded fragments of predicate logic. *J. of Philosophical Logic*, 27(3):217–274, 1998.

[Ben83] J. van Benthem. *The logic of time*. Reidel, 1983.

[Bla93] P. Blackburn. Nominal tense logic. *Notre Dame J. of Formal Logic*, 14:56–83, 1993.

[Bla94] P. Blackburn. Tense, temporal reference, and tense logic. *J. of Semantics*, 11:83–101, 1994.

[Bla98] P. Blackburn. Internalizing labeled deduction. Technical Report CLAUS-Report 102, Computerlinguistik, Universität des Saarlandes, 1998. `http://www.coli.uni-sb.de/cl/claus`.

[BMV97] D. Basin, S. Matthews, and L. Viganò. Labelled propositional modal logics: theory and practice. *J. of Logic and Computation*, 7:685–717, 1997.

[BS93] P. Blackburn and E. Spaan. A modal perspective on the computational complexity of attribute value grammar. *J. of Logic, Language and Information*, pages 129–169, 1993.

[BS95] P. Blackburn and J. Seligman. Hybrid languages. *J. of Logic, Language and Information*, 4(3):251–272, 1995. Special issue on decompositions of first-order logic.

[BT98] P. Blackburn and M. Tzakova. Hybridizing concept languages. *Annals of Mathematics and Artificial Intelligence*, 24:23–49, 1998.

[BT99] P. Blackburn and M. Tzakova. Hybrid languages and temporal logics. *Logic J. of the IGPL*, 7(1):27–54, 1999.

[3] However the extension to $\mathcal{ALCO}$@ would have an impact on the optimization techniques for tableaux calculi developed in the description logic community.

[Bul70] R. Bull. An approach to tense logic. *Theoria*, 36:282–300, 1970.

[Chl86] B. Chlebus. Domino-tiling games. *J. Comput. System Sci.*, 32(3):374–392, 1986.

[De 95] G. De Giacomo. *Decidability of class-based knowledge representation formalisms.* PhD thesis, Università di Roma "La Sapienza", 1995.

[Dem99] S. Demri. Sequent calculi for nominal tense logics: a step towards mechanization? In N. Murray, editor, *Conference on Tableaux Calculi and Related Methods (TABLEAUX), Saratoga Springs, USA*, volume 1617 of *LNAI*, pages 140–154. Springer Verlag, 1999.

[DG99] S. Demri and R. Goré. Cut-free display calculi for nominal tense logics. In N. Murray, editor, *Conference on Tableaux Calculi and Related Methods (TABLEAUX), Saratoga Springs, USA*, volume 1617 of *LNAI*, pages 155–170. Springer Verlag, 1999.

[FL79] M. Fisher and R. Ladner. Propositional dynamic logic of regular programs. *J. of Computer and System Sciences*, 18(2):194–211, 1979.

[Gab96] D. Gabbay. *Labelled Deductive Systems.* Clarendon Press, Oxford, 1996.

[Gor96] V. Goranko. Hierarchies of modal and temporal logics with reference pointers. *J. of Logic, Language and Information*, 5(1):1–24, 1996.

[GW99] E. Grädel and I. Walukiewicz. Guarded fixed point logic. In *Proc. of 14th IEEE Symposium on Logic in Computer Science LICS'99, Trento*, 1999.

[Has91] P. Hasle. Building a temporal logic for natural language understanding with the HOL-system. In C. Brown and G. Koch, editors, *Natural Language Understanding and Logic Programming, III*, pages 91–109. Elsevier Science Publishers, 1991.

[HM92] J. Halpern and Y. Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial Intelligence*, 54:319–379, 1992.

[HS86] J. Halpern and Y. Shoham. A propositional modal logic of time intervals. In *Proc. of the First IEEE Symposium on Logic in Computer Science, Cambridge, Massachusetts*, pages 279–292. Computer Society Press, Washington, 1986.

[Lad77] R. Ladner. The computational complexity of provability in systems of modal propositional logic. *SIAM J. of Computing*, 6(3):467–480, 1977.

[ON80] H. Ono and A. Nakamura. On the size of refutation Kripke models for some linear modal and tense logics. *Studia Logica*, 34:325–333, 1980.

[Pri67] A. Prior. *Past, Present and Future.* Oxford University Press, 1967.

[PT91] S. Passy and T. Tinchev. An essay in combinatory dynamic logic. *Information and Computation*, 93:263–332, 1991.

[Rea94] M. Reape. A feature value logic. In C. Rupp, M. Rosner, and R. Johnson, editors, *Constraints, Language and Computation*, Synthese Language Library, pages 77–110. Academic Press, 1994.

[Rij92] M. de Rijke. The modal logic of inequality. *J. of Symbolic Logic*, 57:566–584, 1992.

[SC85] A. Sistla and E. Clarke. Complexity of propositional linear temporal logics. *J. of the ACM*, 32(3):733–749, 1985.

[Sch91] K. Schild. A correspondence theory for terminological logics. In *Proc. of the 12th IJCAI*, pages 466–471, 1991.

[Sel97] J. Seligman. The logic of correct description. In M. de Rijke, editor, *Advances in Intensional Logic*, pages 107–135. Kluwer, 1997.

[Spa93a] E. Spaan. *Complexity of modal logics.* PhD thesis, ILLC. University of Amsterdam, 1993.

[Spa93b] E. Spaan. The complexity of propositional tense logics. In *Diamonds and defaults (Amsterdam, 1990/1991)*, pages 287–307. Kluwer Acad. Publ., Dordrecht, 1993.

[SSS91] M. Schmidt-Schauss and G. Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48:1–26, 1991.

[Tza99] M. Tzakova. Tableaux calculi for hybrid logics. In N. Murray, editor, *Conference on Tableaux Calculi and Related Methods (TABLEAUX), Saratoga Springs, USA*, volume 1617 of *LNAI*, pages 278–292. Springer Verlag, 1999.

[Ven90] Y. Venema. Expressiveness and completeness of an interval tense logic. *Notre Dame J. Formal Logic*, 31(4):529–547, 1990.

[Ven93] Y. Venema. Derivation rules as anti-axioms in modal logic. *J. of Symbolic Logic*, 58:1003–1034, 1993.