The Complexity of Hybrid Logics over Equivalence Relations

Martin Mundhenk Thomas Schneider
Institut für Informatik, Friedrich-Schiller-Universität, Jena, Germany
{mundhenk,schneider}@cs.uni-jena.de

Abstract

This paper examines and classifies the computational complexity of model checking and satisfiability for hybrid logics over frames with equivalence relations. The considered languages contain all possible combinations of the downarrow binder, the existential binder, the satisfaction operator, and the global modality, ranging from the minimal hybrid language to very expressive languages. For model checking, we separate polynomial-time solvable from PSPACE-complete cases, and for satisfiability, we exhibit cases complete for NP, PSpace, NExp, and even N2Exp. Our analysis includes the versions without atomic propositions of all these languages.

1 Introduction

The quintessence of this paper is the following statement.

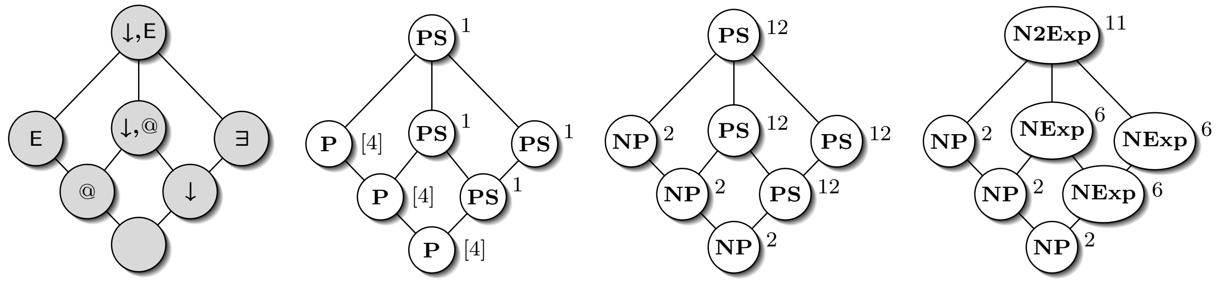
> *Although highly expressive hybrid languages can be tamed by restricting the class of frames, even* very *restricted frame classes have high and different levels of complexity.*

Hybrid logics are powerful and well-behaved extensions of modal logic. However, their expressive power often claims a high price in terms of computational costs: Satisfiability for the language with the "downarrow binder" ↓ is undecidable [1]. Facing this drawback, it is natural to ask for restrictions under which decidability can be restored. One way is to restrict the syntax, for instance by disallowing certain combinations of ↓ and the □ modality, which was examined in [12]. Another way is to restrict the semantics by considering specific frame classes over which ↓ is not as expressive as over the class of all frames. A successful "taming" (i.e., decidability for satisfiability) of the ↓ language has been established for frames of bounded width in [12], and for transitive and complete frames in [9]. Furthermore, over linear frames, where ↓ alone is useless, decidability has been shown for extensions of the ↓ language in [5].

The starting point for our considerations is the NExp-completeness result for satisfiability of the ↓ language over complete frames from [9]. What happens if we enrich the language and allow for slightly more general frames? We examine model checking and satisfiability for hybrid languages with and without propositional variables for each possible combination of ↓, ∃ (a binder stronger than ↓), the satisfaction operator @, and the stronger "somewhere" modality E over frames whose accessibility relation is an equivalence relation (ER frames for short). All these combinations are shown in Figure 1 (a). The results cover a spectrum from polynomial time up to nondeterministic doubly exponential time and thus exhibit the lack of robustness of languages.

The model-checking part of this paper mainly consists of consequences or refinements of results from [4] (where the complexity of model checking for hybrid languages over arbitrary frames has been classified into polynomial-time computable and polynomial-space complete cases). In contrast, our satisfiability part contains new and technically involved results for highly expressive binder languages. The interesting point about those results is that adding the @ operator to the ↓ language or replacing ↓ by the stronger ∃ binder does *not* change complexity, while adding E causes a whole exponential "jump" (from NExp-completeness to N2Exp-completeness). As we will show, this jump is due to two circumstances. First, the logic with ↓ and E lacks the exponential-size model property with respect to frames with equivalence relations. This is because this language is expressive enough to enforce models of doubly exponential size. Second, we can encode a N2Exp-complete version of the the classical bounded tiling problem in these large models.

Our results are visualised in Figure 1 (b)–(d), where the nodes of the diagrams correspond to the languages given in part (a). The abbreviations $\mathcal{HL}$ and $\mathcal{PHL}$ stand for the full and pure (without



(a) hierarchy of all $\mathcal{HL}(\cdot)$ (b) $[\mathcal{P}]\mathcal{HL}(\cdot)$-ER-MC (c) $\mathcal{PHL}(\cdot)$-ER-SAT (d) $\mathcal{HL}(\cdot)$-ER-SAT

Figure 1: A hierarchy of hybrid languages and an overview of their complexity.

atomic propositions) language, respectively. The abbreviations in the nodes denote complexities: P for polynomial-time computable, and the rest for completeness with respect to NP, PSpace, NExp, and N2Exp. Each result is marked with the number of the respective theorem or a reference to its origin.

This paper is organised as follows. In Section 2, we begin with basic concepts and notations of hybrid logic, complexity theory, and tilings. Sections 3 and 4 contain our results for model checking and satisfiability, respectively. Due to space limitations, we have moved easier proofs and few parts of more sophisticated proofs into a technical report, see [8]. We conclude in Section 5.

2 Preliminaries

2.1 Hybrid Logic

Hybrid languages are extensions of the modal language allowing for explicit references to states. Here we introduce the languages relevant for our work. The definitions and notations are taken from [1, 2].

Syntax. Let PROP be a countable set of *propositional atoms*, NOM be a countable set of *nominals*, SVAR be a countable set of *state variables*, and ATOM = PROP ∪ NOM ∪ SVAR. It is common practice to denote propositional atoms by $p, q, \ldots$, nominals by $i, j, \ldots$, and state variables by $x, y, \ldots$ The *full hybrid language* $\mathcal{HL}(\downarrow, \exists, @, \mathsf{E})$ is the set of all formulae of the form $\varphi ::= a \mid \neg\varphi \mid \varphi \wedge \varphi' \mid \Diamond\varphi \mid {\downarrow}x.\varphi \mid \exists\varphi \mid @_t\varphi \mid \mathsf{E}\varphi$, where $a \in$ ATOM, $t \in$ NOM ∪ SVAR, and $x \in$ SVAR. We use the well-known abbreviations $\vee$, $\rightarrow$, $\leftrightarrow$, $\top$ ("true"), and $\bot$ ("false"), as well as $\Box\varphi = \neg\Diamond\neg\varphi$, $\forall\varphi = \neg\exists\neg\varphi$, and $\mathsf{A}\varphi = \neg\mathsf{E}\neg\varphi$. Whenever we leave ↓, @, or E out of the hybrid language, we omit the according symbol from $\mathcal{HL}(\cdot)$.

A hybrid formula is called *pure* if it contains no propositional atoms; *nominal-free* if it contains no nominals; and a *sentence* if it contains no free state variables. (*Free* and *bound* are defined as usual; the only binding operators here are ↓, ∃.)

Semantics for $\mathcal{HL}(\downarrow, \exists, @, \mathsf{E})$ is defined in terms of *Kripke models.* A Kripke model is a triple $\mathcal{M} = (M, R, V)$, where M is a nonempty set of *states*, $R \subseteq M \times M$ is a binary relation — the *accessibility relation* —, and $V : \text{PROP} \rightarrow \mathfrak{P}(M)$ is a function — the *valuation function.* The structure $\mathcal{F} = (M, R)$ is called a *frame.*

A *hybrid model* is a Kripke model with the valuation function V extended to PROP ∪ NOM, where for all $i \in$ NOM, $\#V(i) = 1$. Whenever it is clear from the context, we will omit "hybrid" when referring to models. In order to evaluate ↓- and ∃-formulae, an *assignment* $g : \text{SVAR} \rightarrow M$ *for* $\mathcal{M}$ is necessary. Given an assignment g, a state variable x and a state m, *an x-variant* g^x_m *of* g is defined by $g^x_m(x) = m$ and $g^x_m(x') = g(x')$ for all $x' \neq x$. For any atom a, let $[V, g](a) = \{g(a)\}$ if $a \in$ SVAR, and $V(a)$, otherwise. Given a model $\mathcal{M} = (M, R, V)$, an assignment g, and a state $m \in M$, the satisfaction relation for hybrid formulae is defined by

$$\begin{aligned}
\mathcal{M}, g, m &\Vdash a && \text{iff} \;\; m \in [V, g](a),\;\; a \in \text{ATOM}, \\
\mathcal{M}, g, m &\Vdash \neg\varphi && \text{iff} \;\; \mathcal{M}, g, m \nVdash \varphi, \\
\mathcal{M}, g, m &\Vdash \varphi \wedge \psi && \text{iff} \;\; \mathcal{M}, g, m \Vdash \varphi \;\&\; \mathcal{M}, g, m \Vdash \psi, \\
\mathcal{M}, g, m &\Vdash \Diamond\varphi && \text{iff} \;\; \text{for some } n \in M\text{:}\;\; mRn \;\&\; \mathcal{M}, g, n \Vdash \varphi, \\
\mathcal{M}, g, m &\Vdash {\downarrow}x.\varphi && \text{iff} \;\; \mathcal{M}, g^x_m, m \Vdash \varphi,
\end{aligned}$$

$$
\begin{aligned}
\mathcal{M}, g, m \Vdash \exists x.\varphi \quad & \text{iff} \quad \text{for some } n \in M\colon\ \mathcal{M}, g^x_n, m \Vdash \varphi, \\
\mathcal{M}, g, m \Vdash @_t\varphi \quad & \text{iff} \quad \mathcal{M}, g, n \Vdash \varphi, \text{ where } [V,g](t) = \{n\}, \\
\mathcal{M}, g, m \Vdash \mathsf{E}\varphi \quad & \text{iff} \quad \text{for some } n \in M\colon\ \mathcal{M}, g, n \Vdash \varphi.
\end{aligned}
$$

A formula φ is *satisfiable* if there exist a model $\mathcal{M} = (M, R, V)$, an assignment g for $\mathcal{M}$, and a state $m \in M$, such that $\mathcal{M}, g, m \Vdash \varphi$.

The operators $\downarrow$ and $\exists$ are called *binders*; @ and E are sometimes informally called *jumping operators.* There are certain dependencies between these four operators. First, $\downarrow$ can be expressed using $\exists$: $\downarrow x.\varphi$ is equivalent to $\exists x.(x \wedge \varphi)$. Second, $\exists$ can be expressed using $\downarrow$ and E: $\exists x.\varphi$ is equivalent to $\downarrow y.\mathsf{E}\downarrow x.\mathsf{E}(y \wedge \varphi)$. Third, E can be expressed using $\exists$ and @: $\mathsf{E}\varphi$ is equivalent to $\exists x.(@_x\varphi)$. Fourth, @ can be expressed using E: $@_x\varphi$ is equivalent to $\mathsf{E}(x \wedge \varphi)$. In these formulae, x and y are state variables. Only in the last case can x stand for a nominal, too.

Because of these dependencies, arbitrary combinations of the operators $\downarrow, \exists, @, \mathsf{E}$ result in seven different hybrid languages: $\mathcal{HL}$, $\mathcal{HL}(@)$, $\mathcal{HL}(\mathsf{E})$, $\mathcal{HL}(\downarrow)$, $\mathcal{HL}(\downarrow, @)$, $\mathcal{HL}(\exists)$, and $\mathcal{HL}(\downarrow, \mathsf{E})$. The inclusion hierarchy of these languages is given in Figure 1(a). All other combinations coincide with one of these languages. The *pure fragment* of $\mathcal{HL}(X)$ is denoted by $\mathcal{PHL}(X)$.

Frame Classes; Satisfiability and Model Checking Problems. Let $\mathcal{M} = (M, R, V)$ be a hybrid model with the underlying frame $\mathcal{F} = (M, R)$. If we require the accessibility relation to have certain properties, we restrict the class of relevant frames. Two frame classes are important for this paper. The class of *complete frames* is determined by the restriction $R = M \times M$, and the class of *ER frames* is the class of all frames with equivalence relations. In the latter case, call each equivalence class of $\mathcal{F}$ a *cluster.*

For any hybrid language $\mathcal{HL}(\cdot)$ and any frame class $\mathfrak{F}$, the *satisfiability problem* $\mathcal{HL}(\cdot)$-$\mathfrak{F}$-SAT is defined as follows: Given a formula $\varphi \in \mathcal{HL}(\cdot)$, do there exist a hybrid model $\mathcal{M}$ based on a frame from $\mathfrak{F}$, an assignment g for $\mathcal{M}$, and a state $m \in M$ such that $\mathcal{M}, g, m \Vdash \varphi$? The *model checking problem* $\mathcal{HL}(\cdot)$-$\mathfrak{F}$-MC is defined as follows: Given a formula $\varphi \in \mathcal{HL}(\cdot)$, a hybrid model $\mathcal{M}$ based on a frame from $\mathfrak{F}$, and an assignment g for $\mathcal{M}$, does $\mathcal{M}, g, m \Vdash \varphi$ hold for some state m from $\mathcal{M}$? (If no binder is in the considered language, the assignment g can be left out of either formulation.)

For example, the satisfiability problem over complete frames for the $\downarrow$ language is $\mathcal{HL}(\downarrow)$-compl-SAT, while the model checking problem over ER frames for the $\exists$,@ language is denoted by $\mathcal{HL}(\exists, @)$-ER-MC.

Bounded Model Properties. A logic $\mathcal{HL}(\cdot)$ is said to have the *$f(n)$-size model property* with respect to some class $\mathfrak{F}$ of frames, for some computable function $f : \mathbb{N} \to \mathbb{N}$, iff each formula $\varphi \in \mathcal{HL}(\cdot)$-$\mathfrak{F}$-SAT is satisfiable in a model from $\mathfrak{F}$ that has at most $f(|\varphi|)$ states. This property is important for proving upper complexity bounds of certain logics.

2.2 Further Basic Concepts

Complexity. We refer to [10] for an introduction into complexity theory. In our classification, we use the complexity classes P and NP ((nondeterministic) polynomial time), PSPACE (polynomial space), NEXP and N2EXP (nondeterministic time $2^{\mathrm{poly}(n)}$ and $2^{2^{\mathrm{poly}(n)}}$, respectively). It is known that PSPACE is closed under nondeterminism, that is, PSPACE = NPSPACE. A PSPACE-complete problem is QSAT, which consists in determining whether a given *Quantified Boolean Formula (QBF)* is valid. QBF are first-order formulae of the form $\mathsf{Q}_1 x_1 \ldots \mathsf{Q}_n x_n \alpha(x_1, \ldots, x_n)$, where each Q_i is either $\exists$ or $\forall$, and $\alpha(x_1, \ldots, x_n)$ is a Boolean formula with only the x_i as free variables.

Domino tiling problems are a helpful tool to establish lower complexity bounds for logics. A *tile* is a unit square, divided into four triangles by its diagonals. A *tile type* is a colouring of these four triangles and cannot be rotated. More formally, a tile type t is a quadruple $t = \big(\mathrm{left}(t), \mathrm{right}(t), \mathrm{top}(t), \mathrm{bot}(t)\big)$ of colours. Given a set T of tile types, a T-tiling of the square with side length n is a complete covering of that square with tiles having types from T, such that each point (x, y) is covered by exactly one tile, adjacent tiles have the same colour at their common edges, and the outer border of the square is coloured white. Formally, a T-tiling of the $n \times n$ square is a function $\tau : \{0, 1, \ldots, n-1\} \times \{0, 1, \ldots, n-1\} \to T$ satisfying the following condition for all $(x, y) \in \{0, 1, \ldots, n-1\} \times \{0, 1, \ldots, n-1\}$.

$$
\mathrm{right}\big(\tau(x,y)\big) = \mathrm{left}\big(\tau(x+1,y)\big) \quad \& \quad \mathrm{top}\big(\tau(x,y)\big) = \mathrm{bot}\big(\tau(x,y+1)\big) \tag{1}
$$

$$
\mathrm{bot}\big(\tau(x,0)\big) = \mathrm{top}\big(\tau(x,n-1)\big) = \mathrm{left}\big(\tau(0,y)\big) = \mathrm{right}\big(\tau(n-1,y)\big) = \mathrm{white} \tag{2}
$$

The *square tiling problem* denotes the following question. Given a finite set T of tile types and a string 1^n of n consecutive 1s, is there a T-tiling of the square with side length n? This problem is NP-complete as was shown in [11]. The proof technique used in [11] translates Turing machine computations into tilings and is very robust, such that simple variants of the square tiling problem can straightforwardly be shown to be complete for larger complexity classes. We will consider the following variant, which we call the 2^{2^n}-Tiling problem. Given a finite set T of tile types and a string 1^n, is there a T-tiling of the $2^{2^n} \times 2^{2^n}$ square? This problem is N2EXP-complete.

3 Model Checking

Franceschet and de Rijke [4] investigated model checking for hybrid logics with the $\downarrow$ and $\exists$ binders. Their hardness results hold for the pure nominal-free fragments of these languages. With a slight modification of their proof technique, it is possible to establish the same lower bound over ER frames. The proof of the following theorem is given in [8].

Theorem 1 *Let X be $\{\downarrow\}$, $\{\downarrow, @\}$, $\{\exists\}$, or $\{\downarrow, \mathsf{E}\}$. Then $\mathcal{PHL}(X)$-ER-MC and $\mathcal{HL}(X)$-ER-MC are* PSPACE*-complete.*

4 Satisfiability

4.1 The languages without binders

We show NP-completeness of satisfiability for all pure and non-pure languages without binders, which is the same complexity as for modal logic over ER frames [7]. The lower bound is almost trivial, and the upper bound is due to the $\mathcal{O}(n^2)$-size model property, which is established by a generalisation of the selection procedure given in [7]. The proof of the following theorem is given in [8].

Theorem 2 *Let X be $\emptyset$, $\{@\}$, or $\{\mathsf{E}\}$. Then $\mathcal{HL}(X)$-ER-SAT and $\mathcal{PHL}(X)$-ER-SAT are* NP*-complete.*

4.2 The languages with binders and without E

We consider the languages $\mathcal{HL}(\downarrow)$, $\mathcal{HL}(\downarrow, @)$, and $\mathcal{HL}(\exists)$ and show that satisfiability is NEXP-complete (Theorem 6). Using the hierarchy of the languages, it suffices to prove that $\mathcal{HL}(\downarrow)$-ER-SAT is NEXP-hard (Lemma 3 — which follows immediately from [9, Theorem 3]), and that $\mathcal{HL}(\downarrow, @)$-ER-SAT and $\mathcal{HL}(\exists)$-ER-SAT are in NEXP (Lemmas 4 and 5).

Lemma 3 *$\mathcal{HL}(\downarrow)$-ER-SAT is* NEXP*-hard.*

Lemma 4 *$\mathcal{HL}(\downarrow, @)$-ER-SAT is in* NEXP.

Proof. It suffices to reduce $\mathcal{HL}(\downarrow, @)$-ER-SAT to $\mathcal{HL}(\downarrow)$-compl-SAT, which is in NEXP [9]. This reduction will rely on two basic observations. First, it suffices to consider sentences only, because free state variables can be replaced by nominals without affecting satisfiability. Second, a satisfying ER model for an $\mathcal{HL}(\downarrow, @)$ sentence φ consists w.l.o.g. of not more clusters than there are nominals in φ plus one.

To put the last observation more formally, let φ be an $\mathcal{HL}(\downarrow, @)$ sentence with nominals $i_1, \ldots, i_n$. If φ is satisfied in a state m of a model $\mathcal{M}$, then φ is satisfied in the restriction of $\mathcal{M}$ to the clusters that contain m and all $V(i_k)$. This is so because other clusters are not accessible by means of $\Diamond$ or @.

Hence we can assume w.l.o.g. that a satisfying model for φ consists of at most $n+1$ clusters. Clearly $n \leqslant |\varphi|$. Such a model can be transformed into a model consisting of one "new" cluster that is the union of all these "old" clusters. The old clusters can be distinguished by fresh atomic propositions $c_0, \ldots, c_n$, which help simulate $\Diamond$ and @ using only $\Diamond$. This simulation is captured by the following translation from $\mathcal{HL}(\downarrow, @)$ to $\mathcal{HL}(\downarrow)$ using a fresh state variable x.

$$\begin{aligned} a^t &= a, \quad \text{for } a \in \text{ATOM} & (\Diamond\varphi)^t &= \downarrow x.\Diamond\Big(\textstyle\bigwedge_{k=0}^{n}\big(c_k \leftrightarrow \Box(x \to c_k)\big) \wedge \varphi^t\Big) \\ (\neg\varphi)^t &= \neg\varphi^t & (@_v\varphi)^t &= \Diamond(v \wedge \varphi^t) \\ (\varphi \wedge \psi)^t &= \varphi^t \wedge \psi^t & (\downarrow v.\varphi)^t &= \downarrow v.\varphi^t \end{aligned}$$

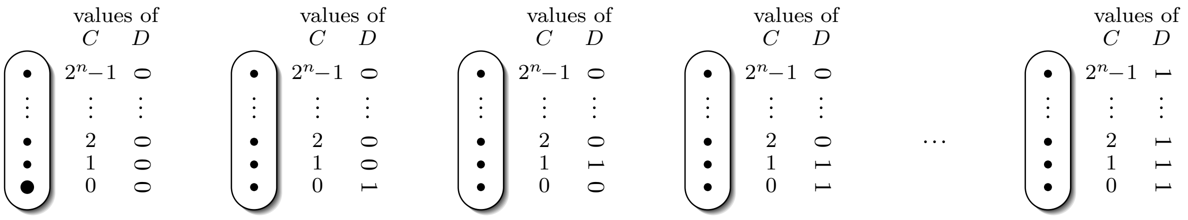


Figure 2: The behaviour of the counters C and D in an ER model.

With the help of the translation $(\cdot)^t$, we define the reduction function $f : \mathcal{HL}(\downarrow, @) \to \mathcal{HL}(\downarrow)$ by

$$f(\varphi) = \varphi^t \ \wedge \ c_0 \ \wedge \ \Box \bigvee\nolimits_{k=0}^{n} c_k \ \wedge \ \Box(i_k \to c_k) \ \wedge \ \bigwedge\nolimits_{\substack{k,\ell=0,\dots,n \\ k \neq \ell}} \Big(\Box(c_k \leftrightarrow c_\ell) \vee \Box\big((c_k \to \neg c_\ell) \wedge (c_\ell \to \neg c_k)\big)\Big),$$

where the conjuncts after φ^t express that φ is satisfied in cluster 0; each state of the new cluster belongs to some old cluster; nominal i_k is true in cluster k; and two clusters k, ℓ are either equal or disjoint. It remains to prove that $\varphi \in \mathcal{HL}(\downarrow, @)$-ER-SAT if and only if $f(\varphi) \in \mathcal{HL}(\downarrow)$-compl-SAT, see [8]. ❏

Lemma 5 *$\mathcal{HL}(\exists)$-ER-SAT is in* NExp.

Lemma 5 can be proven analogously, see [8]. From Lemmas 3, 4, and 5 we obtain the complete characterisation of the satisfiability problems for hybrid logics with $\downarrow$ and without E.

Theorem 6 *Let X be $\{\downarrow\}$, $\{\exists\}$, or $\{\downarrow, @\}$. Then $\mathcal{HL}(X)$-ER-SAT is* NExp*-complete.*

4.3 The full language

In contrast to $\mathcal{HL}(\downarrow)$-ER-SAT and $\mathcal{HL}(\downarrow, @)$-ER-SAT, the complexity of $\mathcal{HL}(\downarrow, \mathsf{E})$-ER-SAT is one exponential level higher. The main reason for this property is the fact that small formulae can enforce satisfying models of doubly exponential size. We will show that it is possible, but not quite straightforward, to enforce a tiling in such big models, which establishes N2Exp-hardness. On the other hand, we will prove that each satisfying model for an $\mathcal{HL}(\downarrow, \mathsf{E})$-formula φ can be restricted to a submodel of doubly exponential size that still satisfies φ. This will allow for a guess-and-check procedure running in N2Exp.

Lemma 7 *For each $n \in \mathbb{N}$ there is a formula $\varphi_n \in \mathcal{HL}(\downarrow, \mathsf{E})$ with the following properties.*

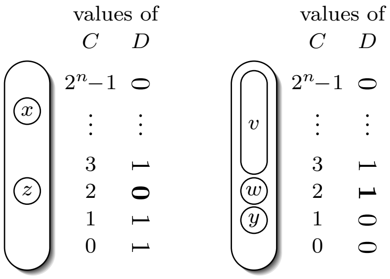
(i) $|\varphi_n| \in \mathcal{O}(n^2)$ *(ii)* $\varphi_n \in \mathcal{HL}(\downarrow, \mathsf{E})$-ER-SAT

(iii) Each satisfying ER model for φ_n has at least 2^{2^n} clusters with 2^n states each.

Proof. In order to enforce a model of the required size, we will proceed in two steps. In the first step, we will implement a counter C that ranges over the values $0, \dots, 2^n - 1$ within each cluster. This will make it possible, for each cluster, to distinguish 2^n states. The counter C is realised by atomic propositions $c_{n-1}, \dots, c_0$ whose truth values, in this order, constitute the binary representation of the value of C at the respective state. (The "truth value" of c_i at the state m is 1 if $m \in V(c_i)$, and 0 otherwise, as usual.)

In the second step we will implement a counter D that ranges over the values $0, \dots, 2^{2^n} - 1$ and distinguishes 2^{2^n} clusters (not states). It will be realised by one atomic proposition d. Given a cluster X, the binary representation of the value of D at X is determined by the truth values of d at the states in X, in the order given by their C-values. Such a doubly exponential counter has been used in [6] to establish lower bounds on the size of certain concepts in Description Logic.

The required behaviour of C and D in a satisfying model for φ_n is visualised in Figure 2, where points and "sausages" represent states and clusters, respectively. The values of C and D in each state are displayed next to it. In the case of C, the shown number determines the truth values of all c_i as described above, and in case of D the given number is the truth value of d. The respective value of the whole counter D becomes readable after turning the D column counterclockwise by 90 degrees. The state with $C = 0$ in the cluster with $D = 0$ shall be the state that satisfies φ_n. It is marked by a larger point.



Name the current state y. Name the state in the x-Cluster with $\neg d$ and lowest possible C-value z. For the state in the y-Cluster with the same C-value as z (which we call w only in this description and in the picture), require three things: (a) d has to hold at w; (b) $\neg d$ has to hold at all states of the y-Cluster with C-value less than the C-value of w; (c) every state v of the y-Cluster with C-value greater than the C-value of w has to agree in d with the states of the x-Cluster that have the same C-value as v.

Figure 3: Incrementation of the D counter.

All these enforcements, of course, will make heavy use of the $\downarrow$ operator combined with E. We will now show how to achieve the required behaviour of C and D. This will be via several formulae whose conjunction results in φ_n. We start with the conjuncts enforcing that each cluster has exactly 2^n states among which every value of C between 0 and 2^n-1 occurs once. In order to keep notation short, we will introduce some abbreviations. First, we would like to refer to specific C-values directly and abbreviate $(C=0) \ = \ \neg c_0 \wedge \ldots \wedge \neg c_{n-1}$ and $(C \neq 2^n-1) \ = \ \neg c_0 \vee \ldots \vee \neg c_{n-1}$. Second, it will be necessary to express that, for some $x \in \mathrm{SVAR}$, the C-value at the current state equals one plus the C-value of the state to which x is bound. (Recall that $@_x\psi$ abbreviates $\mathsf{E}(x \wedge \psi)$.)

$$(C = C_x + 1) \ = \ \bigvee_{k=0}^{n-1} \Big[c_k \wedge @_x \neg c_k \ \wedge \ \bigwedge_{\ell=0}^{k-1} (\neg c_\ell \wedge @_x c_\ell) \ \wedge \ \bigwedge_{\ell=k+1}^{n-1} (c_\ell \leftrightarrow @_x c_\ell) \Big]$$

In addition, we will use analogous shortcuts $C \overset{>}{\underset{<}{=}} C_x$ expressing that the C-value at the current state is less than, equals, or is greater than the C-value of the state to which x is bound. The following conjuncts enforce the required behaviour of each cluster with respect to C.

- ▶ At the state satisfying φ_n, $C=0$ holds. $\quad \mathsf{CZERO}_1 \ = \ (C=0)$
- ▶ In each cluster there is a state with $C=0$. $\quad \mathsf{CZERO}_2 \ = \ \mathsf{A}\Diamond(C=0)$
- ▶ Each cluster has at most one state of each C-value. $\quad \mathsf{CUNIQUE} \ = \ \mathsf{A}{\downarrow}x.\Box\big((C=C_x) \to x\big)$
- ▶ For each state of C-value $c < 2^n-1$, there is a state of C-value $c+1$ in the same cluster.

$$\mathsf{CSUCC} \ = \ \mathsf{A}\big[(C \neq 2^n-1) \to {\downarrow}x.\Diamond(C = C_x + 1)\big]$$

We will now construct the part of φ_n that implements the counter D. This requires expressing that the value of D in the cluster of the current state equals one plus the value of D in the cluster of the state assigned to some state variable x. The appropriate macro is described and illustrated in Figure 3.

$$(D = D_x + 1) \ = \ {\downarrow}y.@_x\Box\,{\downarrow}z.\Big[\big(\neg d \wedge \Box((C < C_z) \to d)\big) \to$$

$$\Big[@_y\Box\Big((C = C_z) \to \big(d \wedge \Box((C < C_z) \to \neg d\big) \wedge \Box\big((C > C_z) \to {\downarrow}v.@_x\Box((C = C_v) \to (d \leftrightarrow @_v d))\big)\big)\Big)\Big]\Big]$$

We easily obtain the two remaining conjuncts for φ_n.

- ▶ The state satisfying φ_n belongs to a cluster with $D=0$. $\quad \mathsf{DZERO} \ = \ \Box\neg d$
- ▶ For each cluster X of D-value $d < 2^{2^n}-1$, there is a cluster Y of D-value $d+1$.

$$\mathsf{DSUCC} \ = \ \mathsf{A}{\downarrow}x.\big(\Diamond\neg d \to \mathsf{E}(D = D_x + 1)\big)$$

Now let $\varphi_n \ = \ \mathsf{CZERO}_1 \wedge \mathsf{CZERO}_2 \wedge \mathsf{CUNIQUE} \wedge \mathsf{CSUCC} \wedge \mathsf{DZERO} \wedge \mathsf{DSUCC}$. Since each of the above abbreviations is of at most quadratic size and they do not occur nested in φ_n, Part (i) of the theorem is satisfied. For (ii), it is easy to see that the following ER model satisfies φ_n at the state $\langle 0,0 \rangle$ under any assignment. Use $\mathrm{bin}_i(n)$ to denote the i-th bit in the binary representation of $n \in \mathbb{N}$, and let $\mathcal{M}^n = (M^n, R^n, V^n)$ as follows.

$$\begin{array}{rclrcl} M^n & = & \{\langle x,y \rangle \mid x,y \in \mathbb{N};\ 0 \leqslant x < 2^{2^n};\ 0 \leqslant y < 2^n\} & V^n(c_i) & = & \{\langle x,y \rangle \mid \mathrm{bin}_i(y) = 1\} \\ R^n & = & \{\big(\langle x_1,y_1 \rangle, \langle x_2,y_2 \rangle\big) \mid x_1 = x_2\} & V^n(d) & = & \{\langle x,y \rangle \mid \mathrm{bin}_y(x) = 1\} \end{array} \qquad (3)$$

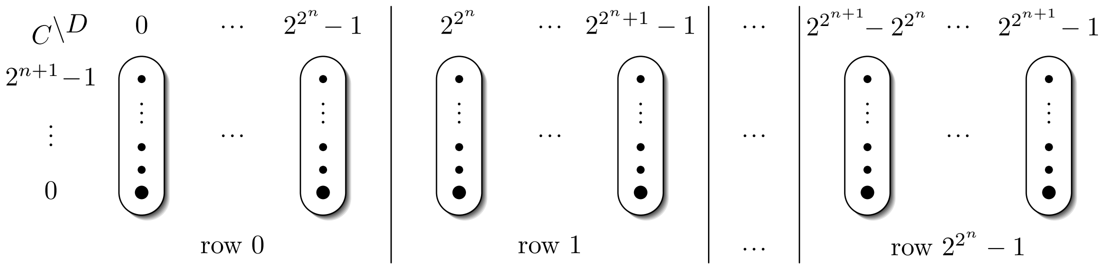


Figure 4: Enforcing a tiling in an ER model of doubly exponential size.

In order to show (iii), let $\mathcal{M}=(M,R,V)$ be an ER model with $m_{0,0}\in M$ and g be an assignment for $\mathcal{M}$ such that $\mathcal{M},g,m_{0,0}\Vdash\varphi_n$. Now the four C-conjuncts enforce that $C=0$ at $m_{0,0}$, and that each cluster of $\mathcal{M}$ contains exactly one state of C-value c for each $c=0,\dots,2^n-1$. Due to DZERO, the D-value of $m_{0,0}$'s cluster equals 0, and DSUCC successively enforces the existence of a cluster of D-value d for each $d=0,\dots,2^{2^n}-1$. (Note that the value of D in each cluster is uniquely determined by $V(d)$ and the order of the cluster's states determined by their C-values.) Hence $\mathcal{M}$ has at least 2^{2^n} clusters with 2^n states each. ❑

Corollary 8 *$\mathcal{HL}(\downarrow,\mathsf{E})$ does not have the $2^{\mathrm{poly}(n)}$-size model property with respect to ER frames.*

Theorem 9 *$\mathcal{HL}(\downarrow,\mathsf{E})$-ER-SAT is* N2EXP*-hard.*

Proof. We will reduce the 2^{2^n}-tiling problem to $\mathcal{HL}(\downarrow,\mathsf{E})$-ER-SAT. The reduction will use the techniques enforcing doubly exponentially large satisfying models from the proof of Lemma 7. In order to encode a tiling for the $2^{2^n}\times 2^{2^n}$-square in an ER model $\mathcal{M}$, we will first enforce that $\mathcal{M}$ has $2^{2^{n+1}}$ clusters with 2^{n+1} states each, using the same construction of counters C and D, but with parameter $n+1$. The tiled square itself will be encoded in the states of C-value 0 of all clusters. Hence row 0 of the square will be in the clusters of D-value $0,\dots,2^{2^n}-1$; row 1 will be in the clusters of D-value $2^{2^n},\dots,2\cdot 2^{2^n}-1$; etc.; see Figure 4. The horizontal adjacencies in the original square can be expressed referring to pairs of clusters with successive D-values. In contrast, for the vertical adjacencies, pairs of clusters whose D-values differ by 2^{2^n} will have to be compared.[1]

For the required reduction, we will show how to transform an instance $\langle T,n\rangle$ of the tiling problem into a formula $\psi_{T,n}$ such that there is a T-tiling of the $2^{2^n}\times 2^{2^n}$-square if and only if $\psi_{T,n}$ is satisfiable. As in the proof of Lemma 7, this formula will consist of several conjuncts. The first of them will be the formula φ_{n+1} from that proof, enforcing the required structure of the model. In order to keep the remaining conjuncts short, we will use the same abbreviations again, but with $n+1$ instead of n. Furthermore, $D=D_x+2^{2^n}$ denotes that the D-value of the current state's cluster equals 2^{2^n} plus the D-value of the cluster containing the state to which x is bound. This abbreviation is defined analogously to the shortcut $D=D_x+1$.

Now we are ready to give the conjuncts that enforce the tiling. Let T be a set of tile types. For each tile type t we will use an atomic proposition t to denote that a tile of type t lies at the respective position.

- At each state with C-value 0 lies exactly one tile. $\quad\mathsf{TILE}\ =\ \mathsf{A}\Big((C=0)\rightarrow\bigvee_{t\in T}\big(t\wedge\bigwedge_{\substack{t'\in T\\ t'\neq t}}\neg t\big)\Big)$

- Tiles match horizontally and vertically.

$$\mathsf{HOR}\ =\ \mathsf{A}\Big[\big((C=0)\wedge\Diamond(\neg c_n\wedge d)\big)\rightarrow{\downarrow}x.\Big(\bigwedge_{t\in T}\rightarrow\mathsf{A}\big(((C=0)\wedge(D=D_x+1))\rightarrow\bigvee_{t'\in\mathrm{RI}(t)}t'\big)\Big)\Big]$$

The $\Diamond$-subformulae require that corresponding position of the current state does not belong to the last column (or row, respectively) of the square. The conjunct VER is defined analogously.

[1] Note that this is not the standard way to encode tilings in models, insofar as not every state of the model corresponds to a position in the grid. However, this modification of the standard way is not new, since it relies on ideas developed by Chlebus in [3] to encode the rectangle tiling problem with exponential parameter into a more intricate version of a bounded tiling problem that he called “High Tiling”.

▶ The borders of the square are white. (The three omitted conjuncts are analogous.)

$$\mathsf{WHITE} \;=\; \mathsf{A}\Big[\Big(\Box(c_n \rightarrow \neg d) \rightarrow \textstyle\bigvee_{\substack{t\in T\\ \mathrm{bot}(t)=\mathrm{white}}} t\Big) \;\wedge\; \ldots \;\wedge\; \ldots \;\wedge\; \ldots$$

Now let $\psi_{T,n} = \varphi_{n+1} \wedge \mathsf{TILE} \wedge \mathsf{HOR} \wedge \mathsf{VER} \wedge \mathsf{WHITE}$. Each conjunct is of size at most $\mathcal{O}(n^2 + |T|^2)$. From their definitions it is clear that $\psi_{T,n}$ can be computed in time polynomial in $n + |T|$. It is straightforward to show that there is a T-tiling of the $2^{2^n} \times 2^{2^n}$-square if and only if $\psi_{T,n} \in \mathcal{HL}(\downarrow, \mathsf{E})$-ER-SAT, see [8]. ❑

We will now establish the corresponding upper bound, showing that the full hybrid language has a doubly exponential size model property over ER frames. This will make it possible to decide satisfiability using a straightforward check-and-guess procedure and involving results for model checking.

Lemma 10 *$\mathcal{HL}(\downarrow, \mathsf{E})$ has the $2^{2^{2n+2}}$-size model property with respect to ER frames.*

Proof. Intuitively, the proof relies on the following considerations: Call the set of propositional variables and nominals that hold at a given state of a model the *type* of this state. Let the *C-type* of a cluster be the set of types of all points of this cluster. If we had no $\downarrow$ in our language, then two states of the same type that belong to the same cluster would not be distinguishable, that is, they would satisfy the same formulae. Even two states of the same type that belong to two different clusters of the same C-type would not be distinguishable. This would enable us to restrict clusters to at most one state per possible type and to restrict a whole satisfying model for some formula φ to at most one cluster per possible C-type without affecting satisfiability of φ.

In the presence of $\downarrow$, this argumentation must be refined and requires a certain amount of technical details. Let φ be a formula of size n and $\mathcal{M} = (M, R, V)$ be a satisfying model for φ. First, there are at most 2^n possible types of states. Since an assignment for $\mathcal{M}$ might bind all state variables occurring in φ to different states of the same type, only up to $n+1$ states of the same type belonging to the same cluster are distinguishable. Hence, it is legitimate to restrict each cluster of $\mathcal{M}$ to at most $n+1$ states of each type in the first step, which leads to an exponential bound in the size of clusters.

In the second step, we modify the notion of a C-type of a cluster X to be the multiset containing as many copies of each type as there are states of this type in X, but not more than $n+1$. It is legitimate, too, to restrict the whole model to at most $n+1$ clusters of each C-type. Since there are at most $(n+2)^{2^n}$ many different C-types, the number of clusters — and, hence, states — of the restricted model is bounded by $2^{2^{\mathcal{O}(n)}}$.

The formal proof of the $2^{2^{2n+2}}$-size model property requires quite some notation. Let $\varphi \in \mathcal{HL}(\downarrow, \mathsf{E})$-ER-SAT be of size n. Then there exist an ER model $\mathcal{M} = (M, R, V)$, an assignment g_0 for $\mathcal{M}$, and a state $m_0 \in \mathcal{M}$ such that $\mathcal{M}, g_0, m_0 \Vdash \varphi$. Let $C_i \subseteq M$, $i \in I$, be all clusters of $\mathcal{M}$, for an appropriate index set I that contains 0, such that $m_0 \in C_0$. Let $x_1, \ldots, x_s$ be all state variables occurring in φ. Analogously, let $a_1, \ldots, a_t$ be all other atoms in φ. Clearly $s, t \leqslant n$. A *φ-type* is a subset of $\{a_1, \ldots, a_t\}$. Let $A_1, \ldots, A_{2^t}$ be an enumeration of all φ-types, such that m_0 is of type A_1. (A state m is of type A_ℓ iff for each $j = 1, \ldots, t$: $(m \in V(a_j) \Leftrightarrow a_j \in A_\ell)$. Furthermore, we will deliberately speak of "(C-)types" instead of "φ-(C-)types" whenever no confusion may arise.) Given a cluster C, we divide it into 2^t "type layers" $C_i^\ell = \{m \in C_i \mid m \text{ is of type } A_\ell\}$.

We define a function $f : I \times \{1, \ldots, 2^t\} \rightarrow \mathfrak{P}(M)$ that assigns a set of states to each pair $\langle i, \ell \rangle$ of a cluster number i and a type number ℓ, such that $f(i, \ell)$ is a subset of C_i. The union of all possible $f(i, \ell)$ will constitute the first restriction of $\mathcal{M}$. The function f is defined as follows, where $\#C_i^\ell$ denotes the number of states in C_i^ℓ. If $\#C_i^\ell \leqslant s+1$, then $f(i, \ell) = C_i^\ell$. Otherwise, $f(i, \ell)$ is some subset of C_i^ℓ of size at most $s+1$ that satisfies the following conditions.

(i) For each $j = 1, \ldots, s$: if $g_0(x_j) \in C_i^\ell$, then $g_0(x_j) \in f(i, \ell)$. (ii) $m_0 \in f(0, 1)$.

Such a subset always exists. For any cluster C_i, let $f(C_i)$ denote the union of all $f(i, \ell)$. Due to the definition of f, $f(C_i) \subseteq C_i$, and $f(C_i)$ has at most $(s+1) \cdot 2^t$ states. We denote the union of all $f(C_i)$ by M'.

After restricting the cluster size, we will restrict the number of the clusters. Let $\mathcal{A}$ be the multiset containing $s+1$ copies of each type A_ℓ. Call each subset of $\mathcal{A}$ a *φ-C-type.* The power set $\mathfrak{P}(\mathcal{A})$ contains $(s+2)^{2^t}$ elements. Let $\mathcal{A}_1, \ldots, \mathcal{A}_{(s+2)^{2^t}}$ be an enumeration of all φ-C-types, such that $f(C_0)$ is of C-type

$\mathcal{A}_1$. (The C-type of a cluster C_i is determined by the number of states of each type in its restriction $f(C_i)$.) We divide M' into $(s+2)^{2^t}$ "C-type layers" $\mathcal{C}_\ell$ being the union of $f(C_i)$ for all C_i of C-type $\mathcal{A}_\ell$.

Now define a second choice function $f' : \{1, \ldots, (s+2)^{2^t}\} \to \mathfrak{P}(M')$ that assigns a set of states to each C-type number such that $f'(\ell)$ is a union of (restricted) clusters. The union of all possible $f'(\ell)$ will constitute the second restriction of $\mathcal{M}$. The function f' is defined as follows. If there are not more than $s+1$ clusters of C-type $\mathcal{A}_\ell$, then $f'(\ell) = \mathcal{C}_\ell$. Otherwise, $f'(\ell)$ is the union of $s+1$ restricted clusters of type $\mathcal{A}_\ell$ satisfying

(iii) $\forall j = 1, \ldots, s$: if $g_0(x_j) \in f(C_i)$ for some C_i of type $\mathcal{A}_\ell$, then $f(C_i) \subseteq f'(\ell)$; (iv) $f(C_0) \subseteq f'(1)$.

Such a subset always exists. Due to the definition of f', each $f'(\ell)$ contains at most $s+1$ restricted clusters and, hence, $(s+1)^2 \cdot 2^t$ states. We now construct a new model $\mathcal{M}'' = (M'', R'', V'')$ from $\mathcal{M}$, where M'' is the union of $f'(\ell)$ for all C-types $\mathcal{A}_\ell$, and R'' and V'' are the restrictions of R and V to M''. Now the following facts about $\mathcal{M}''$ are obvious. It is still an ER model, whose clusters are restrictions of clusters of $\mathcal{M}$. It contains m_0, because $m_0 \in f(C_0) \subseteq f'(1)$. The assignment g_0 is an assignment for $\mathcal{M}''$. Since there are $(s+2)^{2^t}$ C-types, M'' contains $(s+2)^{2^t} \cdot (s+1)^2 \cdot 2^t$ states. This number is limited by $2^{2^{2n+2}}$ because $s, t \leqslant n$.

It remains to show that $\mathcal{M}'', g_0, m_0 \Vdash \varphi$. For this purpose, we make use of an auxiliary statement. This statement uses the concept of agreement in a pair of assignments. We say that two states m and m' from $\mathcal{M}$ agree in two assignments g/g' for $\mathcal{M}$ iff $\{x_k \mid g(x_k) = m\} = \{x_k \mid g'(x_k) = m'\}$. Two clusters C_i and $C_{i'}$ agree in g/g' iff they are of the same C-type, and for each A_ℓ, each $m \in C_i^\ell$, there is some $m' \in C_{i'}^\ell$ that agrees with m in g/g'.

Claim 1. For each subformula ψ of φ; for each two assignments g, g' for $\mathcal{M}$; for each C-type $\mathcal{A}_\ell$; for each two clusters C_i and $C_{i'}$ that agree in g/g'; for each type A_ℓ; and for each $m \in C_i^\ell$ and $m' \in C_{i'}^\ell$ that agree in g/g'; it holds that $\mathcal{M}, g, m \Vdash \psi$ iff $\mathcal{M}, g', m' \Vdash \psi$.

Now the required fact $\mathcal{M}'', g_0, m_0 \Vdash \varphi$ is a consequence of the following claim.

Claim 2. For each subformula ψ of φ, for each $m \in M''$, for each assignment g for $\mathcal{M}$", it holds that $\mathcal{M}, g, m \Vdash \psi$ iff $\mathcal{M}'', g, m \Vdash \psi$.

Claims 1 and 2 are proven in [8]. ❑

The following theorem is an almost immediate consequence of Lemma 10 and [4, Theorem 4.5]. For details of the proof, see [8].

Theorem 11 $\mathcal{HL}(\downarrow, \mathsf{E})$-ER-SAT *is* N2EXP-*complete.*

4.4 Pure languages with binders

Satisfiability for all pure languages with binders is PSPACE-complete. The lower bound is due to an easy reduction from QSAT similarly to that for the model checking problem in Theorem 1. The upper bound uses a polynomial-size model property that is obtained in a similar manner as the $2^{2^{2n+2}}$-size model property for $\mathcal{HL}(\downarrow, \mathsf{E})$ in Lemma 10. Note the following subtle difference in argumentation. While the $2^{2^{2n+2}}$-size model property of $\mathcal{HL}(\downarrow, \mathsf{E})$ implies an N2EXP upper bound for satisfiability, the polynomial-size model property of a binder language does not imply an NP upper bound for satisfiability. The reason becomes clear if we recall the general complexity results for model checking over arbitrary frames from [4]: In the presence of binders, this problem is PSPACE-complete, but an upper time bound is $\mathcal{O}(|\varphi| \cdot |M|^{2|\varphi|})$. If the model is large compared to the formula, as in the case of $\mathcal{HL}(\downarrow, \mathsf{E})$, then the factor $|\varphi|$ in the exponent is unimportant. In the case of a polynomial-size model property, however, the upper time bound for model checking only yields an exponential time bound for the whole guess-and-check algorithm deciding satisfiability. The proof of Theorem 12 is given in [8].

Theorem 12 *Let X be $\{\downarrow\}$, $\{\downarrow, @\}$, $\{\exists\}$, or $\{\downarrow, \mathsf{E}\}$. Then $\mathcal{PHL}(X)$*-ER-SAT *is* PSPACE-*complete.*

5 Conclusion

We have completely classified the computational complexity of model checking and satisfiability over ER frames for all hybrid languages shown in Figure 1 (a). In detail, we have established the following results.

Model checking is in polynomial time for each binder-free language, and PSpace-complete in the cases with binders. In all seven cases, the pure fragment has the same complexity.

Satisfiability is NP-complete for all binder-free cases, whether pure or with propositional variables. This is the same complexity as for modal logic over equivalence relations [7]. For the four languages with binders, there is a significant gap in complexity between the pure and non-pure cases. The former are PSpace-complete, while the latter are NExp-complete if E is not in the language, and even N2Exp-complete with E. As for the last case, we have established a $2^{2^{2n+2}}$-size model property for $\mathcal{HL}(\downarrow, \mathsf{E})$ with respect to ER frames, and we have disproven a $2^{\mathrm{poly}(n)}$-size model property.

The scope of our results is slightly larger than stated in Theorems 1, 6, 11, and 12, in the sense that all these statements hold as well for the nominal-free fragments of all sentences of the respective languages $\mathcal{HL}(\cdot)$ and $\mathcal{PHL}(\cdot)$. This is due to the fact that neither nominals nor free state variables occur in the particular reductions used for the lower bounds. (Except for the case of Lemma 3, to be precise. However, the lower NExp bound for $\mathcal{HL}(\downarrow)$-compl-SAT does hold for nominal-free sentences as well, because nominals and free state variables can be simulated in complete frames using bound state variables.) The only case in which the lower bound does not carry over to the pure fragment is that of satisfiability for binder-free languages (see Theorem 2).

References

[1] Areces, C., P. Blackburn and M. Marx, *A road-map on complexity for hybrid logics*, in: *Proc. of the 13th CSL, 1999*, LNCS **1683** (1999), pp. 307–321.

[2] Areces, C., P. Blackburn and M. Marx, *The computational complexity of hybrid temporal logics*, Logic Journal of the IGPL **8** (2000), pp. 653–679.

[3] Chlebus, B. S., *Domino-tiling games.*, J. Comput. Syst. Sci. **32** (1986), pp. 374–392.

[4] Franceschet, M. and M. de Rijke, *Model checking for hybrid logics (with an application to semistructured data)*, Journal of Applied Logic **4** (2006), pp. 279–304.

[5] Franceschet, M., M. de Rijke and B.-H. Schlingloff, *Hybrid logics on linear structures: Expressivity and complexity*, in: *Proc. of the 10th TIME, 2003* (2003), pp. 166–173.

[6] Ghilardi, S., C. Lutz and F. Wolter, *Did I damage my ontology? A case for conservative extensions in description logics*, in: P. Doherty, J. Mylopoulos and C. Welty, editors, *Proc. 10th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR'06)* (2006), pp. 187–197.

[7] Ladner, R. E., *The computational complexity of provability in systems of modal propositional logic*, SIAM Journal on Computing **6** (1977), pp. 467–480.

[8] Mundhenk, M. and T. Schneider, *The complexity of hybrid logics over equivalence relations*, Technical Report 07-02, Reports on Computer Science, Friedrich-Schiller-Universität Jena (2007). URL `http://www.minet.uni-jena.de/Math-Net/reports/reports.html`

[9] Mundhenk, M., T. Schneider, T. Schwentick and V. Weber, *Complexity of hybrid logics over transitive frames*, in: H. Schlingloff, editor, *M4M*, Informatik-Berichte **194** (2005), pp. 62–78.

[10] Papadimitriou, C. H., "Computational Complexity," Addison-Wesley, 1994.

[11] Savelsbergh, M. and P. van Emde Boas, *Bounded tiling, an alternative to satisfiability*, in: G. Wechsung, editor, *2nd Frege Conference*, Mathematische Forschung **20** (1984), pp. 354–363.

[12] ten Cate, B. and M. Franceschet, *On the complexity of hybrid logics with binders*, in: *Proc. of the 19th CSL, 2005*, LNCS **3634** (2005), pp. 339–354.