On the Complexity of Elementary Modal Logics*  

Edith Hemaspaandra and Henning Schnoor  
Department of Computer Science, Rochester Institute of Technology, Rochester, NY 14623, U.S.A.  
eh@cs.rit.edu, hs@cs.rit.edu

Abstract. Modal logics are widely used in computer science. The complexity of modal satisfiability problems has been investigated since the 1970s, usually proving results on a case-by-case basis. We prove a very general classification for a wide class of relevant logics: Many important subclasses of modal logics can be obtained by restricting the allowed models with first-order Horn formulas. We show that the satisfiability problem for each of these logics is either NP-complete or PSPACE-hard, and exhibit a simple classification criterion. Further, we prove matching PSPACE upper bounds for many of the PSPACE-hard logics.

1 Introduction

Modal logics have proven to be a valuable tool in mathematics and computer science. The traditional uni-modal logic enriches the propositional language with the operator ♦, where ♦φ is interpreted as φ possibly holds. The usual semantics interpret modal formulas over graphs, where ♦φ means “there is a successor world where φ is true.” In addition to their mathematical interest, modal logics are widely used in practical applications: In artificial intelligence, modal logic is used to model the knowledge and beliefs of an agent, see e.g. [BZ05]. Modal logics also can be applied in cryptographic and other protocols [FHJ02,CDF03,HMT88,LR86]. For many specific applications, there exist tailor-made variants of modal logics [BG04].

Due to the vast number of applications, complexity issues for modal logics are very relevant, and have been examined since Ladner’s seminal work [Lad77]. Depending on the application, modal logics with different properties are studied. For example, one might want the formula φ ⇒ ♦φ to be an axiom—if something is true, then it should be considered possible. Or ♦♦φ ⇒ ♦φ—if it is possible that φ is possible, then φ itself should be possible. Classical results [Sah73] show that there is a close correspondence between modal logics defined by axioms and logics obtained by restricting the class of considered graphs. Requiring the axioms mentioned above corresponds to restricting the classes of graphs to those which are reflexive or transitive, respectively. Determining the complexity of a given modal logic, defined either by the class of considered graphs or via a modal axiom system, has been an active line of research since Ladner’s results. In particular, the complexity classes NP and PSPACE have been at the center of attention.

Most complexity results have been on a case-by-case basis, proving results for individual logics both for standard modal logics and variations like temporal or hybrid logics [HM92,Ngu05,SC85]. Examples of more general results include Halpern and Régó’s proof that logics including the negative introspection axiom, which corresponds to the Euclidean graph property, have an NP-complete satisfiability problem [HR07]. In [SP06], Schröder and Pattinson show a way to prove PSPACE upper bounds for modal logics defined by modal axioms of modal depth 1. In [Lad77], Ladner proved PSPACE-hardness for all logics for which

* Supported in part by NSF grants CCR-0311021 and IIS-0713061, a Friedrich Wilhelm Bessel Research Award, and the DAAD postdoc program.
reflexive and transitive graphs are admissible models. In [Spa93], Hemaspaandra showed that all normal logics extending S4.3 have an NP-complete satisfiability problem, and work on the Guarded Fragment has shown that some classes of modal logics can be seen as a decidable fragment of first-order logic [AvBN98].

While these results give hardness or upper bounds for classes of logics, they do not provide a full case distinction identifying all "easy" or "hard" cases in the considered class. We achieve such a result: For a large class of modal logics containing many important representatives, we identify all cases which have an NP-complete satisfiability problem, and show that the satisfiability problem for all other non-trivial logics in that class is PSPACE-hard. Hence these problems avoid the infinitely many complexity classes between NP and PSPACE, many of which have natural complete problems arising from logical questions. To our knowledge, such a general result has not been achieved before.

To describe the considered class of modal logics, note that many relevant properties of modal models can be expressed by first-order formulas: A graph is transitive if its edge-relation $R$ satisfies the clause $\forall xyz (xRy \land yRz \implies xRz)$ and symmetric if it satisfies $\forall xy (xRy \implies yRx)$. Many other graph properties can be defined using similar formulas, where the presence of a certain pattern of edges in the graph forces the existence of another. Analogously to propositional logic, we call conjunctions of such clauses universal Horn formulas. Many relevant logics can be defined in this way: All examples form [Lad77] fall into this category, as well as logics over Euclidean graphs.

We study the following problem: Given a universal Horn formula $\hat{\psi}$, what is the complexity of the modal satisfiability problem over the class of graphs defined by $\hat{\psi}$?

The main results of this paper are the following: First, we identify all cases which give a satisfiability problem solvable in NP (which then for every nontrivial logic is NP-complete), and show that all other cases are PSPACE-hard. Second, we prove a generalization of a "tree-like model property," and use it to obtain PSPACE upper bounds for a large class of logics. As a corollary, we prove that Ladner’s classic hardness result is “optimal” in the class of logics defined by universal Horn formulas. A further corollary is that in the universal Horn class, all logics whose satisfiability problem is not PSPACE-hard already have the “polynomial-size model property,” which is only one of several known ways to prove NP upper bounds for modal logics.

Various work was done on restricting the syntax of the modal formulas by restricting the propositional operators [BHSS06], the nesting degree and number of variables [Hal95] or considering modal formulas in Horn form [CL94]. While these results are about restricting the syntax of the modal formulas, the current work studies different semantics of modal logics, where the semantics are specified by Horn formulas.

The organization of the paper is as follows: In Section 2, we introduce terminology and generalize classic complexity results. Section 3 then establishes techniques to restrict the size of models for modal formulas, which are important tools for the NP-membership later. Section 4 contains the main results of the paper about universal Horn formulas. After introducing them in Section 4.1 and proving their relationship to homomorphisms in Section 4.2, we prove NP-results for special cases of Horn formulas in Sections 4.3 and 4.4. Using these results, Section 4.5 then proves our main dichotomy result, which is Corollary 4.29. The remainder of the paper establishes PSPACE upper bounds for many of the PSPACE-hard logics. An important tool for these proofs is introduced in Section 4.6, where we show a tree-like model property for all PSPACE-hard logics defined by universal Horn formulas. Section 4.7 contains our PSPACE-algorithm, which generalizes many previously known algorithms for modal logics. Finally, Section 4.8 obtains a series of corollaries, proving the above-mentioned optimality result for Ladner’s hardness condition, and exhibiting a number
of cases to which our PSPACE-algorithm can be applied. The paper closes with a summary and open questions in Section 5.

2 Preliminaries

2.1 Basic concepts and notation

Modal logic is an extension of propositional logic. A modal formula is a propositional formula using variables, the usual logical symbols $\land, \lor, \neg$, and a unary operator $\Diamond$. (A dual operator $\Box$ is often considered as well, this can be regarded as abbreviation for $\neg \Diamond \neg$.) A model for a modal formula is a set of connected “worlds” with individual propositional assignments. To be precise, a frame is a directed graph $G = (W, R)$, where the vertices in $W$ are called “worlds,” and an edge $(u, v) \in R$ is interpreted as $v$ is “considered possible” from $u$. A model $M = (G, X, \pi)$ consists of a frame $G = (W, R)$, a set $X$ of propositional variables and a function $\pi$ assigning each variable $x \in X$ a subset of $W$, the set of worlds in which $x$ is true. We say the model $M$ is based on the frame $(W, R)$. If $F$ is a class of frames, then a model is an $F$-model if it is based on a frame in $F$. With $|M|$ we denote the number of worlds in the model $M$.

For a world $w \in W$, we define when a formula $\phi$ is satisfied at $w$ in $M$ (written $M, w \models \phi$.) If $\phi$ is a variable $x$, then $M, w \models \phi$ if and only if $w \in \pi(x)$. As usual, $M, w \models \phi_1 \land \phi_2$ if and only if $M, w \models \phi_1$ and $M, w \models \phi_2$, and $M, w \models \neg \phi$ iff $M, w \not\models \phi$. For the modal operator, $M, w \models \Diamond \phi$ if and only if there is a world $w' \in W$ such that $(w, w') \in R$ and $M, w' \models \phi$.

| logic name | graph property | formula definition |
|------------|----------------|-------------------|
| K          | All graphs     | $K(\varphi_{taut})$ |
| T          | reflexive      | $K(\varphi_{refl})$ |
| B          | symmetric      | $K(\varphi_{symm})$ |
| K4         | transitive graphs | $K(\varphi_{trans})$ |
| S4         | transitive and reflexive | $K(\varphi_{trans} \land \varphi_{refl})$ |
| S5         | equivalence relations | $K(\varphi_{trans} \land \varphi_{refl} \land \varphi_{symm})$ |

Table 1. Common modal logics

We now describe a way to define classes $F$ of frames by propositional formulas. The frame language is the first-order language containing (in addition to the propositional operators $\land, \lor, \neg$) the binary relation $R$. The relation $R$ is interpreted as the edge relation in a graph. Semantics are defined in the obvious way, for example, a graph satisfies the formula $\varphi_{trans} := \forall x, y, z (xRy) \land (yRz) \Rightarrow (xRz)$ if and only if it is transitive. In order to separate modal formulas from first-order formulas, we use $\hat{\varphi}$ to denote the latter, i.e., $\hat{\varphi}$ is a first-order formula, while $\varphi$ is a modal formula.

A modal logic usually is defined as the set of the formulas provable in it. Since a formula is satisfiable iff its negation is not provable, we can define a logic by the set of formulas satisfiable in it. For a first-order formula $\hat{\varphi}$ over the frame language, we define the logic $K(\hat{\varphi})$ as the logic in which a modal formula $\phi$ is satisfiable if and only if there is a model $M$ and a world $w \in M$ such that the frame which $M$ is based on satisfies the first-order formula $\hat{\varphi}$ (we simply write $M \models \hat{\varphi}$ for this), and $M, w \not\models \phi$. Such a logic is called elementary. In the case that $\hat{\varphi}$ is a universal formula (i.e., every variable in $\hat{\varphi}$ is universally quantified at the beginning of the formula), we call these logics universal elementary. In this way, many
of the classic examples of modal logics can be expressed: In addition to the formula \(\hat{\varphi}_{\text{trans}}\) defined above, let \(\hat{\varphi}_{\text{refl}} := \forall w (w R w)\), and let \(\hat{\varphi}_{\text{symm}} := \forall x, y (x R y) \implies (y R x)\). Finally, let \(\hat{\varphi}_{\text{taut}}\) be some tautology over the frame language, for example let \(\hat{\varphi}_{\text{taut}} := \forall x (x R x) \implies (x R x)\). Table 1 introduces some common modal logics and how they can be expressed in our framework. For a formula \(\hat{\varphi}\) over the frame language, we consider the following problem:

**Problem:** K(\(\hat{\varphi}\))-SAT  
**Input:** A modal formula \(\phi\)  
**Question:** Is \(\phi\) satisfiable in a model based on a frame satisfying \(\hat{\varphi}\)?

As an example, the problem K(\(\hat{\varphi}_{\text{trans}}\))-SAT is the problem to decide if a given modal formula can be satisfied in a transitive frame, and therefore is the same as the satisfiability problem for the logic K4. It is important to note that in the problem K(\(\hat{\varphi}\))-SAT, regard the formula \(\hat{\varphi}\) is fixed. It is also interesting to study the uniform version of the problem, where we are given a first-order formula \(\hat{\varphi}\) over the frame language and a modal formula \(\psi\), and the goal is to determine whether there exists a graph satisfying both. This problem obviously is PSPACE-hard (this easily follows from Ladner’s Theorem 2.4, in fact, the problem is undecidable). In this paper, we study the complexity behavior of fixed modal logics.

When interested in complexity results for modal logic, the property of having “small models” is often crucial, as these lead to a satisfiability problem in NP, as long as the class of frames considered is reasonably well-behaved.

**Definition 2.1.** A modal logic KL has the polynomial-size model property, if there is a polynomial \(p\), such that for every KL-satisfiable formula \(\phi\), there is a KL-model \(M\) and a world \(w \in M\) such that \(M, w \models \phi\), and \(|M| \leq p(|\phi|)\).

The following standard observation is the basis of our NP-containment proofs:

**Proposition 2.2.** Let \(\hat{\varphi}\) be a first-order formula over the frame language, such that K(\(\hat{\varphi}\)) has the polynomial-size model property. Then K(\(\hat{\varphi}\))-SAT \(\in\) NP.

**Proof.** This easily follows from the literature, since for a given graph and a fixed first-order sentence \(\hat{\varphi}\), it can be checked in polynomial time if the graph satisfies \(\hat{\varphi}\). Also, it can be verified in polynomial time if a model satisfies a modal formula. Hence, the obvious guess-and-verify approach works for NP-containment. \(\square\)

Since modal logic is an extension of propositional logic, the satisfiability problem for every non-trivial modal logic is NP-hard. Therefore, proving the polynomial-size model property yields an optimal upper complexity bound for the satisfiability problem for modal logics.

### 2.2 Ladner’s Theorem and Applications

In the seminal paper [Lad77], Ladner showed PSPACE-containment and PSPACE-hardness for a variety of modal logics. In particular, he proved that the satisfiability problem for any logic between K and S4 is PSPACE-hard. In order to state Ladner’s result, we introduce the concept of extensions of a logic, and how it relates to modal logics defined by first-order formulas.

For a modal logic KL, an extension of KL is a modal logic KL’ such that every formula which is valid (a tautology) in KL is also valid in KL’, or equivalently such that every formula that is KL’-satisfiable is also KL-satisfiable. As an example, every logic that we consider is an extension of K, and S4 is an extension of K4. In the case of elementary logics, this is related to an implication of the corresponding first-order-formulas. In the following, when we say that a formula \(\hat{\varphi}\) over the frame language implies a formula \(\hat{\psi}\) over the frame language, then we mean that every graph which is a model of \(\hat{\varphi}\) also satisfies \(\hat{\psi}\).
Proposition 2.3. Let $\varphi$ and $\psi$ be first-order formulas over the frame language, and let $K(\varphi)$ and $K(\psi)$ be the corresponding elementary modal logics. If $\varphi$ implies $\psi$, then $K(\varphi)$ is an extension of $K(\psi)$.

Proof. Let $\varphi$ be a modal formula which is valid in $K(\psi)$. Then $\neg\varphi$ is not $K(\psi)$-satisfiable. Now assume that $\neg\varphi$ is $K(\psi)$-satisfiable. Then there exists a model $M$ and a world $w \in M$ such that $M, w \models \neg\varphi$ and $M$ is a $K(\varphi)$-model, i.e., $M \models \varphi$. Since $\varphi$ implies $\psi$, we know that $M \models \psi$, and therefore $M$ is a $K(\psi)$-model. Therefore, $\neg\varphi$ is $K(\psi)$-satisfiable, a contradiction. Therefore we know that $\neg\varphi$ is not $K(\varphi)$-satisfiable, and hence $\varphi$ is valid in $K(\varphi)$. Therefore, it follows that $K(\varphi)$ is an extension of $K(\psi)$. 

Note that the converse of Proposition 2.3 does not hold. For example, consider the formulas $\varphi_1 = \exists x (xRx)$, and $\varphi_2 = \forall x (xRx) \vee (xRx)$. Then $\varphi_2$ is a tautology and $\varphi_1$ is not, in particular, we know that $\varphi_2$ does not imply $\varphi_1$. But the logics $K(\varphi_1)$ and $K(\varphi_2)$ are easily seen to be identical (and both identical to $K$). In particular, they are extensions of each other.

Ladner’s main result can be stated as follows:

Theorem 2.4 ([Lad77]).

1. The satisfiability problems for the logics $K$, $K4$, and $S4$ are PSPACE-complete, and $S5$-SAT is NP-complete.
2. Let $KL$ be a modal logic such that $S4$ is an extension of $KL$. Then $KL$-SAT is PSPACE-hard.

Ladner’s proof for Theorem 2.4 shows some additional results, which we will give as a series of corollaries.

The construction from Ladner’s proof can be modified to prove the following result as well:

Corollary 2.5. Let $\varphi$ be a formula over the frame language which is satisfied in every symmetric tree or in every reflexive and symmetric tree. Then $K(\varphi)$-SAT is PSPACE-hard.

Even though the proof is just a minor variation of Ladner’s proof, we give the entire construction for completeness. We mention where the adjustments for the symmetric case are.

Proof. The result follows from a slight modification of Ladner’s proof for the hardness result in Theorem 2.4. We follow the presentation of [BdRV01], where Ladner’s theorem can be found as Theorem 6.50.

The result follows from a slight modification of Ladner’s proof for the hardness result in Theorem 2.4. We follow the presentation of [BdRV01], where Ladner’s theorem can be found as Theorem 6.50.

The proof shows a reduction from the evaluation problem for quantified Boolean formulas, $QBF$. The main strategy is to create from a quantified formula $\chi$ a modal formula $\varphi$ such that in any satisfying model for $\varphi$, a complete “quantifier tree” for the quantifier block of $\chi$ can be found, i.e., a tree where for each existentially quantified variable, one value is chosen, and for universally quantified variables, both alternatives true and false are evaluated.

Let $\chi = Q_1 p_1 \ldots Q_m p_m \theta(p_1, \ldots, p_m)$ be a quantified Boolean formula, where $Q_i \in \{\forall, \exists\}$, and $\theta$ is a propositional formula. From this we construct a modal formula $\phi$, in which variables $p_1, \ldots, p_m$ and $q_1, \ldots, q_m$ appear. The $p_i$ correspond directly to the variables of the propositional formula, and the $q_i$ mark the level of the node in the modal model: As mentioned, a model $M$ for the formula $\phi$ is essentially a quantifier tree, and hence each node has a unique level in $M$. The construction will ensure that (up to the depth of the model that we care about) $q_i$ is true in a state if and only if the state is on level $i$ in the model.
The variables $p_i$ express the values of the variables in the quantifier tree, where the value of $p_i$ is only regarded as “defined” from level $i$ onwards (obviously, these variables also have values in other levels, but the values in these levels are not of interest to us).

For constructing the formula which forces the quantifier tree, we first introduce two macros. The macro $B_i$ requires the model to “split” at level $i$:

$$B_i := q_i \rightarrow \diamond(q_{i+1} \land p_{i+1}) \land \diamond(q_{i+1} \land \overline{p_{i+1}}).$$

The effect of the formula $B_i$ is that if it is required to be true in level $i$, then each node in level $i$ must have two different successors in the next level, setting the variable $p_{i+1}$ to true in one of them, and to false in the other. Therefore we can use this macro to force the “branching” of the quantifier tree in the levels corresponding to universal variables: if $p_{i+1}$ is a universally quantified variable, then the macro $B_i$ ensures that both possible truth values for $p_{i+1}$ are evaluated.

When forcing the quantifier tree, we also need to ensure that truth values for the variables $p_i$ are properly propagated to “lower levels” in the tree. For this, we use the following macro (and this is the only point in which our construction differs from the proof given in [BdRV01]):

$$S_i := ((p_i \rightarrow \square((q_{i+1} \lor \cdots \lor q_m) \rightarrow p_i)) \land (\overline{p_i} \rightarrow \square((q_{i+1} \lor \cdots \lor q_m) \rightarrow \overline{p_i}))).$$

This formula forces the truth value of $p_i$ to be propagated “down” the tree from those levels on where we actually regard the value of $p_i$ as set, i.e., from the level $i$ on. Note that a node $v$ can only have a successor in which $(q_{i+1} \lor \cdots \lor q_m)$ holds if its own level is at least $i$, and hence we restrict the values of $p_i$ in exactly those parts of the tree where it is regarded as defined.

We now give the construction of the formula $\varphi$, which as mentioned is identical to the one used in Ladner’s proof, with the exception that our macro $S_i$ is different. The formula $\varphi$ is the conjunction of the following formulas (here, $\square^m\psi$ is an abbreviation for a $\underbrace{\square\cdots\square}_m\psi$, and $\square^m\psi$ is a shorthand for $\psi \land \square^1\psi \land \square^2\psi \land \cdots \land \square^m\psi$):

1. $q_0$
2. $\square^m(q_i \rightarrow \bigwedge_{j \neq j} \overline{p_j})$ \quad $(0 \leq i \leq m)$
3. $\square^m(q_i \rightarrow \diamond q_{i+1})$ \quad $(0 \leq i < m)$
4. $\bigwedge_{i \mid q_i = \forall} \square^m B_i$
5. $\square^m S_i$ \quad $(0 \leq i < m)$
6. $\square^m(q_m \rightarrow \theta)$

The construction works as follows: With formula (i), we give the start of the model and define the world $w$ in which $\varphi$ is satisfied to have the level 0. Formula (ii) requires each node which is reachable in at most $m$ steps from $w$ to have a well-defined level (or none). Formulas (iiia) and (iiib) require each level $i$ to be followed by level $i + 1$, and in the case that the $i$th quantifier is $\forall$, formula (iiib) requires the corresponding branching of the quantifier tree. Formula (iv) then forces the truth values of the $p_i$ to be “sent down” the tree, as described earlier. Formula (v) finally requires that the propositional formula $\theta$ is true for all possible truth assignments to $p_1, \ldots, p_m$ generated by the quantifier tree.

Ladner’s construction for the PSPACE upper bound for the logic $K$ also reveals the following. The main idea behind this corollary is that any model for some modal formula can be transformed into a strict tree by “unrolling.” A much more general version of this result will be proven later as Theorem 4.33.
Corollary 2.6. A modal formula is K-satisfiable if and only if it can be satisfied in a strict tree.

Corollary 2.6 shows that we only need to apply our formulas to trees—as long as our first order formula does not “say anything” about trees, the generated logic is the same as K, although these logics do not necessarily have the same set of models.

Corollary 2.7. Let \( \hat{\varphi} \) be a first-order formula over the frame language such that \( \hat{\varphi} \) is satisfied in every strict tree. Then the satisfiability problem for K(\( \hat{\varphi} \)) is PSPACE-complete.

Proof. This follows immediately from Corollary 2.6 and Theorem 2.4, since every modal formula is K-satisfiable if and only it is K(\( \hat{\varphi} \))-satisfiable. \( \square \)

For our complete classification in Theorem 4.17, we need a hardness result which is a slight variation of Ladner’s hardness result for all logics between K and S4. However, the proof is merely a closer inspection of Ladner’s construction.

In addition to the already mentioned graph properties, we define a generalization of transitivity. For a natural number \( k \), we say that a graph \( G \) is \( k \)-transitive, if every pair of vertices \( (u, v) \) in \( G \) such that there is a \( k \)-step path from \( u \) to \( v \) in \( G \) is connected with an edge. It is easy to see that a graph is transitive if and only if it is 2-transitive, and a transitive graph is also \( k \)-transitive for every \( k \in \mathbb{N} \). For a set \( S \subseteq \mathbb{N} \), we say that a graph is \( S \)-transitive if it is \( k \)-transitive for every \( k \in S \).

Theorem 2.8. Let \( \hat{\psi} \) be a first-order formula over the frame language such that one of the following cases applies:

- \( \hat{\psi} \) is satisfied in every strict tree,
- \( \hat{\psi} \) is satisfied in every reflexive tree,
- there is a set \( S \subseteq \mathbb{N} \) such that \( \hat{\psi} \) is satisfied in every \( S \)-transitive tree,
- \( \hat{\psi} \) is satisfied in every symmetric tree,
- \( \hat{\psi} \) is satisfied in every tree which is both reflexive and symmetric,
- there is a set \( S \subseteq \mathbb{N} \) such that \( \hat{\psi} \) is satisfied in every tree which is both reflexive and \( S \)-transitive.

Then K(\( \hat{\psi} \))-SAT is PSPACE-hard.

Note that for the cases not including symmetry, the result “almost” follows directly from Ladner’s Theorem 2.4. It does not follow directly, since we only require that our first-order formulas are satisfied in \( S \)-transitive trees, but nothing about arbitrary \( S \)-transitive graphs. Nevertheless, the result for this case follows directly from Ladner’s proof.

Proof. First consider the cases in which symmetry does not occur. Now the result follows directly from Ladner’s proof: Ladner proves PSPACE-hardness with a reduction from QBF. From a quantified Boolean formula \( \chi \), he constructs a modal formula \( \phi \) such that the following holds:

1. If \( \chi \) is true, then there is a modal model \( M \) and a world \( w \in M \) such that \( M \) is a strict tree, and \( M, w \models \phi \). Moreover, adding any number of reflexive or transitive edges in \( M \) preserves the fact that \( M, w \models \phi \).
2. If \( \chi \) is false, then \( \phi \) is not K-satisfiable.
It is immediate that this reduction also proves the desired hardness result for $K(\hat{\psi})$:

First assume that $\chi$ is true. Then, by the above, the model $M, w$ satisfies $\phi$, and is a tree. If we close this model under the reflexive and/or $S$-transitive closures, then by the above this still is a model for $\phi$. Since $\hat{\psi}$ is satisfied in every reflexive and/or $S$-transitive tree, this is a $K(\hat{\psi})$-model for $\phi$, and hence $\phi$ is $K(\hat{\psi})$-satisfiable. On the other hand, if $\chi$ is false, then $\phi$ is not $K$-satisfiable, and therefore not $K(\hat{\psi})$-satisfiable.

Now the cases involving symmetry follow from Corollary 2.5. $\square$

As this section indicated, trees are an important subclass of modal models. And in fact, a good intuition to read this paper is to always think of the graphs we deal with as “near-trees,” i.e., trees with additional edges.

### 3 About Modal Models

NP-results in this paper are shown with the explicit construction of small models for a given formula. We therefore introduce some notation on graphs. For a graph $G$, the set of vertices of $G$ is denoted with $\text{vertices}(G)$, and $\text{edges}(G)$ is the set of its edges. As usual, a homomorphism from a graph $G_1$ to a graph $G_2$ is a function preserving the edge relation. A strict tree is a tree in the usual sense, i.e., a directed, acyclic, connected graph which has a root $w$ from which all other vertices can be reached. We now define notation to describe paths in graphs. Note that we often identify modal models and their frames, when the propositional assignments are clear from the context or not important for our arguments.

**Definition 3.1.** Let $G$ be a graph, $w, v \in G$ vertices, and $i \in \mathbb{N}$. We write $G \models w \xrightarrow{i} v$ if in $G$, there is a path of length $i$ from $w$ to $v$. Additionally, $G \models w \xrightarrow{0} w$ for all $w \in G$. We also say that $w$ is a $i$-step predecessor of $v$, and $v$ is a $i$-step successor of $w$ in $G$, if $G \models w \xrightarrow{i} v$. The maximal depth of $w \in G$ is defined as $\text{maxdepth}^G(w) := \max \{i \mid \exists w' \in G, G \models w' \xrightarrow{i} w\}$. Similarly, the maximal height of $w \in G$ is $\text{maxheight}^G(w) := \max \{i \mid \exists w' \in G, G \models w \xrightarrow{i} w'\}$.

Note that the maximal depth and maximal height of nodes can be (countably) infinite, even in finite graphs. The next definition is a restriction on graphs which is very natural for modal logics: for deciding whether $M, w \models \phi$ holds for some modal $M$, a world $w \in M$, and a modal formula $\phi$, it is obvious that only the worlds which are reachable from $w$ are important.

**Definition 3.2.** Let $G$ be a graph, and let $w \in G$. The graph $G_w$ is obtained from $G$ by restricting $G$ to the worlds which can be reached from $w$.

#### 3.1 Invariants

Proving the polynomial-size model property for some logic is usually done starting with an arbitrary model for a given modal formula and building a smaller model out of it, which still satisfies the modal formula. However, we also need to ensure that the new model still satisfies the conditions of the logic under consideration. Therefore, we need results that allow us to perform modifications on our models and leave the modal and the first-order properties invariant. The first result in this way concerns the first-order aspect of the frames for universal formulas:
Theorem 3.3. Let $G$ and $G'$ be graphs, and let $n$ be a natural number. The following conditions are equivalent:

1. Every universal first-order formula (with at most $n$ variables) satisfied by $G'$ is also satisfied by $G$.
2. Every existential first-order formula (with at most $n$ variables) satisfied by $G$ is also satisfied by $G'$.
3. For each finite $V \subseteq G$ (such that $|V| \leq n$), there exists a functions $f : V \rightarrow G'$ such that for $u,v \in V$, it holds that $uRv$ iff $f(u)R'f(v)$.

Proof. $1 \rightarrow 2$ Let $\exists x_1 \ldots \exists x_n \hat{\varphi}(x_1, \ldots, x_n)$ be an existential first-order formula over the frame language which holds in $G$, and assume that it does not hold in $G'$. In this case, the negation of the formula holds in $G'$, i.e., $G' \models \exists x_1 \ldots \exists x_n \neg \varphi(x_1, \ldots, x_n)$. This is equivalent to $G' \models \forall x_1 \ldots \forall x_n \varphi(x_1, \ldots, x_n)$, which is a universal first-order formula over the frame language. Hence by the prerequisites, we know that $G \models \forall x_1 \ldots \forall x_n \varphi(x_1, \ldots, x_n)$, which is a contradiction.

$2 \rightarrow 1$ Analogously to the above.

$2 \rightarrow 3$ We construct a function $f : V \rightarrow G'$ with the desired properties. Let $|V| = n$, and let $V = \{x_1, \ldots, x_n\}$. We construct an existential first-order formula

$$\hat{\psi} = \exists x_1, \ldots, \exists x_n \bigwedge_{(i,j) \in R} x_iRx_j \land \bigwedge_{(i,j) \notin R} \neg x_iRx_j.$$  

Obviously, $\hat{\psi}$ has $n$ variables, and $G$ obviously is a model for $\hat{\psi}$, it follows that $G' \models \hat{\psi}$. Therefore, there are $x'_1, \ldots, x'_n$ such that $(x_i, x_j) \in R$ if and only if $(x'_i, x'_j) \in R'$. Define $f(x_i) := x'_i$. This function obviously meets the criteria: Let $(x_i, x_j) \in R$. Then $x_iRx_j$ is a clause in $\hat{\psi}$. Therefore for the values $x'_i$ and $x'_j$ chosen by the existential quantifiers, $(x'_i, x'_j) \in R$ must hold. Since $f(x_i) = x'_i$ and $f(x_j) = x'_j$, the claim follows. For the condition $R'(x_i, x_j)$, the proof is the same.

$3 \rightarrow 2$ Let such a function $f$ exist for every $V \subseteq G$ with $|V| \leq n$. Let $\hat{\psi} := \exists x_1, \ldots, \exists x_n \hat{\varphi}$ be a first-order existential formula, let $n$ be the number of variables in $\hat{\psi}$, and let $G \models \hat{\psi}$. We show that $G' \models \hat{\psi}$. Since $G \models \hat{\psi}$, there are $x_1, \ldots, x_n \in G$ such that $\varphi(x_1, \ldots, x_n)$ holds in $G$. This implies that $\varphi(f(x_1), \ldots, f(x_n))$ holds in $G'$. Therefore, $G' \models \hat{\psi}$. \qed

Note that while the function $f$ required to exist in the conditions of the above theorem shares some properties with an isomorphism, it is not required to be injective. The following is an important special case, which immediately follows from this observation and Theorem 3.3:

Proposition 3.4. Let $G$ be a graph, $\hat{\varphi}$ a universal first-order formula over the frame language such that $G \models \hat{\varphi}$. Then for every subgraph $G'$ of $G$, it holds that $G' \models \hat{\varphi}$.

For a modal model, a restriction of the model is a restriction of the graph, where the propositional assignment for the remaining worlds is unchanged. We now consider restrictions which are “compatible” with the modal properties of the formulas in question. The following lemma describes a standard way to reduce the number “relevant” of successors to worlds in models. This is an application of the more general idea of bounded morphisms, which we will encounter in Section 4.6. For a modal formula $\phi$, $\text{sf}(\phi)$ denotes the set of its subformulas. With $\text{md}(\phi)$, we denote the modal depth of a formula $\phi$, i.e., the maximal nesting degree of the modal operator $\Box$ in $\phi$. 

[1] Analogously to the above.
Lemma 3.5. Let $\phi$ be a modal formula, and let $M, w \models \phi$. Let $M'$ be a restriction of $M$ such that the following holds:

1. $w \in M'$,
2. for all $u \in M'$, and all $\psi \in \text{sf}(\phi)$ such that $M, u \models \Diamond \psi$, there is some $v \in M'$ such that $(u, v)$ is an edge in $M$, and $M, v \models \psi$.

Then $M', w \models \phi$.

Lemma 3.5 immediately follows from the following Lemma. The version stated in Lemma 3.5 is the one we almost exclusively use, hence we stated this simpler version explicitly. We now prove a slightly more general result, which also takes into account that for a modal formula, worlds which are not reachable on a path with at most the length of the modal depth of the formula, are irrelevant.

Lemma 3.6. Let $\phi$ be a modal formula, and let $M, w \models \phi$. Let $M'$ be a restriction of $M$ such that the following holds:

1. $w \in M'$,
2. for all $u \in M'$, and all $\psi \in \text{sf}(\phi)$, such that $M, u \models \Diamond \psi$ and there exists an $i \in \mathbb{N}$ such that $M \models w \xrightarrow{i} u$ and $1 + i + \text{md}(\psi) \leq \text{md}(\phi)$, there is some $v \in M'$ such that $(u, v)$ is an edge in $M$, and $M, v \models \psi$.

Then $M', w \models \phi$.

Proof. We show the following claim: Let $\chi \in \text{sf}(\phi), i \in \mathbb{N}$, and $u \in M'$ such that $M \models w \xrightarrow{i} u$ and $i + \text{md}(\chi) \leq \text{md}(\phi)$, then $M, u \models \chi$ if and only if $M', u \models \chi$. For $\chi = \phi$, $u = w$, and $i = 0$, this implies the Lemma, since $w$ is an element of $M'$ by definition.

We show the claim by induction on $\chi$. If $\chi$ is a variable, then this holds trivially, since $M'$ is a restriction of $M$ and therefore, propositional assignments are not changed. The induction step for propositional operators is trivial. Therefore, assume that $\chi = \Diamond \psi$ for some $\psi \in \text{sf}(\phi)$, such that the claim holds for $\psi$. Now let $u, i$ meet the prerequisites of the claim, i.e., let $i + \text{md}(\chi) \leq \text{md}(\phi)$, and let $M \models w \xrightarrow{i} u$.

First assume that $M, u \models \chi$. Since $\chi = \Diamond \psi$, it follows that $\text{md}(\chi) = \text{md}(\psi) + 1$, and hence $i + 1 + \text{md}(\psi) \leq \text{md}(\phi)$. Since $M, u \models \Diamond \psi$, the prerequisites of the Lemma therefore imply that there is a world $v \in M'$ such that $(u, v)$ is an edge in $M$, and $M, v \models \psi$. Since $M \models w \xrightarrow{i} u$, it follows that $M \models w \xrightarrow{i+1} v$. By the induction hypothesis, we know that $M', v \models \psi$. Since $M'$ is a restriction of $M$, $(u, v)$ is an edge in $M'$ as well, and therefore we conclude that $M', u \models \Diamond \psi$, i.e., $M', u \models \chi$.

For the other direction, assume that $M', u \models \chi$. Therefore, there is a node $v \in M'$ such that $M', v \models \psi$, and $(u, v)$ is an edge in $M'$. Since $M \models w \xrightarrow{i} u$, we know that $M \models w \xrightarrow{i+1} v$ holds as well, and since $\text{md}(\psi) = \text{md}(\chi) - 1$, from the induction hypothesis we conclude that $M, v \models \psi$. Since $(u, v)$ is an edge in $M$ as well, it therefore follows that $M, u \models \Diamond \psi$, i.e., $M, u \models \chi$, concluding the proof.

The following easy proposition shows how Lemma 3.6 can be applied:

Proposition 3.7. Let $\text{KL}$ be a universal elementary logic, let $\phi$ be a modal formula, and let $M, w \models \phi$, where $M$ is a $\text{KL}$-model. Then there is a $\text{KL}$-model $M'$ which is a restriction of $M$, which is rooted at $w$, and where every world can be reached from $w$ in at most $\text{md}(\phi)$ steps, and $M', w \models \phi$.
Proof. The model $M'$ is obtained by simply removing all worlds from $M$ which cannot be reached from $w$ in at most $\text{md}(\phi)$ steps. Due to Proposition 3.4, $M'$ is still a KL-model. We now show that $M', w \models \phi$ holds, by proving that $M'$ satisfies the conditions of Lemma 3.6.

By definition, since $w$ can be reached from $w$ in 0 steps, we know that $w \in M'$. Hence let $u \in M'$, and let $\psi = \text{sf}(\phi)$, and let $M, u \models \Box \psi$, such that $M \models w \xrightarrow{i} u$ for some $i$ such that $1 + i + \text{md}(\psi) \leq \text{md}(\phi)$. Since $M, u \models \Box \psi$, we know that there is a world $v \in M$ such that $M, v \models \psi$, and $(u, v)$ is an edge in $M$. It follows that $M \models w \xrightarrow{i+1} u$. Since $1 + i + \text{md}(\psi) \leq \text{md}(\phi)$, and $\text{md}(\psi) \leq \text{md}(\phi)$, we know that $i + 1 \leq \text{md}(\phi)$, and hence we know that $v$ is an element of $M'$.

Therefore, $M'$ satisfies the conditions of Lemma 3.6, and therefore the lemma implies that $M', w \models \phi$, as claimed. \□

3.2 Restrictions

As mentioned, our NP-containment results are obtained by proving the polysize model property and applying Proposition 2.2. The polynomial models are obtained by restricting arbitrary models to polynomial size. We will now show some restrictions which we can make in any model, showing that we can assume certain parts of the model to be only polynomial in size.

**Lemma 3.8.** Let $c \in \mathbb{N}$. Then for any modal formula $\phi$ and any $M, w \models \phi$, there is a submodel $M'$ of $M$ such that $M', w \models \phi$ and the following holds:

$$\left| \left\{ v \in M' \mid \text{maxdepth}^{M'}(v) < c \right\} \right| \leq (c + 1) \cdot |\phi|^c.$$

**Proof.** For each world $u$ in $M$, let $F_u := \{ \psi \in \text{sf}(\phi) \mid M, u \models \Box \psi \}$. For each $u$, let $W_u$ be a subset of the 1-step successors of $u$ in $M$ such that for every $\psi \in F_u$, there is a world $v \in W_u$ such that $(u, v)$ is an edge in $M$, and $M, v \models \psi$, and $|W_u| \leq |F_u|$. Now define $M_0 := \{ w \}$, and for each $i \in \mathbb{N}$, let $M_{i+1} := \bigcup_{v \in M_i} W_v$. Finally, define $M'$ to be the restriction of $M$ to $\bigcup_{i \in \mathbb{N}} M_i$.

To show that $M', w \models \phi$, we prove that $M'$ satisfies the conditions of Lemma 3.5. Obviously, $M'$ is a restriction of $M$ and $w \in M_0 \subseteq M$. Therefore, let $u$ be a world from $M'$, and let $\psi$ be a subformula of $\phi$ such that $M, u \models \Box \psi$. Since $u \in M'$, there is some $i$ such that $u \in M_i$. Since $M, u \models \Box \psi$, we know that $\psi \in F_u$, and hence there is a world $v \in W_u$ such that $M, v \models \psi$, and $(u, v)$ is an edge in $M$. It follows that $v \in W_u \subseteq M_{i+1} \subseteq M'$, and hence $M'$ satisfies the conditions of Lemma 3.5 as claimed.

Now let

$$A := \left\{ v \in M' \mid \text{maxdepth}^{M'}(v) < c \right\}.$$

It remains to show the cardinality bound for $A$. It is obvious that $A \subseteq \bigcup_{i=0}^{c} M_i$, since for every $i$, every vertex in $M_{i+1}$ has a predecessor in $M_i$, and hence inductively, every vertex in $M_i$ has an $i$-step predecessor in $M'$.

Obviously, $|M_0| = 1$, and $|M_{i+1}| \leq |M_i| \cdot |\text{sf}(\phi)|$. Therefore, $|M_i| \leq |\text{sf}(\phi)|^i$ for all $i \in \mathbb{N}$. Now, due to the above, $|A| \leq \sum_{i=0}^{c} |M_i| \leq (c + 1) \cdot |\text{sf}(\phi)|^c$. Since $|\text{sf}(\phi)| \leq |\phi|$, the claim follows. \□

By construction, the model $M'$ given in the proof of the above lemma is countable. Hence we obtain the following corollary:

**Corollary 3.9.** Let KL be a universal elementary logic. Then every KL-satisfiable formula $\phi$ is satisfiable in a countable KL-model. Moreover, every KL-model satisfying $\phi$ at a node $w$ contains a countable KL-submodel satisfying $\phi$ at the node $w$. 

---

**On the Complexity of Elementary Modal Logics**

11
The following lemma shows that it is sufficient to restrict the number of those vertices in the model which have a minimal height in the graph. In combination with Lemma 3.8, this shows that we only need to be concerned about vertices which have both a certain number of predecessors, and a certain number of successors. We already saw in Proposition 3.7 that we are only interested in rooted graphs. In such graphs, Lemmas 3.8 and 3.10 can be seen as limiting the number of vertices near the “top” or the “bottom” of the model. This is useful, because in graphs satisfying some universal formula, often special cases can occur in these regions of the graph. These lemmas show that we do not need to look too closely at these exceptions.

**Lemma 3.10.** Let \( c \in \mathbb{N} \) be a constant. Then for any modal formula \( \phi \) and any \( M,w \models \phi \), there is a submodel \( M' \) of \( M \) such that \( M',w \models \phi \), and for which the following holds:

\[
|M'| \leq f \left( \left\{ v \in M' \mid \text{maxheight}^{M'}(v) \geq c \right\} \right),
\]

where \( f(n) = (n + 1) \cdot (1 + |\phi|)^c \).

*Proof.* Let \( A := \left\{ v \in M \mid \text{maxheight}^M(v) \geq c \right\} \), i.e., the set of nodes in \( M \) which have a \( c \)-step successor. We now define a sequence of submodels of \( M : \) Let \( M_0 := \emptyset, M_1 := A \cup \{ w \} \), and for \( i \geq 1 \), let \( M_{i+1} \) be defined as follows:

- \( M_i \subseteq M_{i+1} \)
- For every \( u \in M_i \setminus M_{i-1} \), and each \( \psi \in \text{sf} (\phi) \) such that \( M,u \models \Diamond \psi \), add one world \( v \) from \( M \) to \( M_{i+1} \) such that \( (u,v) \) is an edge in \( M \) and \( M,v \models \psi \).

Now let \( M' \) be the restriction of \( M \) to the worlds in \( M_{c+1} \). We show that \( M',w \models \phi \), by showing that it satisfies the conditions of Lemma 3.5. By definition, \( w \in M_1 \subseteq M' \). Hence let \( u \in M', \psi \in \text{sf}(\phi) \), and let \( M,u \models \Diamond \psi \). Since \( u \in M' \), there exists a minimal \( i \) such that \( u \in M_i \). First assume that \( i = c + 1 \). By construction, for every relevant \( j \), every node in \( M_{j+1} \setminus M_j \) has a 1-step predecessor in \( M_j \setminus M_{j-1} \), and hence, inductively, the node \( u \in M_{c+1} \setminus M_c \) is a \( c \)-1-step successor of a node \( x \) in \( M_2 \setminus M_1 \). Since \( M,u \models \Diamond \psi \), we know that \( u \) has a successor in \( M \). This implies that \( x \) has a \( c \)-step successor in \( M \), and hence \( x \in A \), which is a contradiction, since \( x \in M_2 \setminus M_1 \), and \( A \subseteq M_1 \).

Therefore, we know that \( i \leq c \). By construction, there is a world \( v \) in \( M_{c+1} \subseteq M' \) such that \( (u,v) \) is an edge in \( M \), and \( M,v \models \psi \). Therefore, the model \( M' \) satisfies the conditions of Lemma 3.5, and therefore we conclude that \( M',w \models \phi \).

By definition, it holds that \( |M_1| \leq |A| + 1 \), and for \( i \geq 1 \), \( |M_{i+1}| \leq |M_i|(1 + |\text{sf}(\phi)|)^c \). Since \( |M'| = |M_{c+1}| \), this implies that \( |M'| \leq (|A| + 1) \cdot (1 + |\text{sf}(\phi)|)^c \leq (|A| + 1) \cdot (1 + |\phi|)^c \), as claimed. \( \Box \)

The main purpose of Lemmas 3.8 and 3.10 is the following: If for a modal logic \( \text{KL} \), there is a constant \( c \), such that every \( \text{KL} \)-satisfiable formula has a model in which we can restrict the number of nodes which have both a \( c \)-step predecessor and a \( c \)-step successor, then Lemmas 3.8 and 3.10 can be used to show the polynomial model property for \( \text{KL} \). This idea plays a crucial role in the proof of our main NP-containment result, Theorem 4.13, and is formalized in the following corollary:

**Corollary 3.11.** Let \( \text{KL} \) be a universal elementary modal logic such that there exists a constant \( c \in \mathbb{N} \) such that there is a polynomial \( p \) such that for all \( \text{KL} \)-satisfiable formulas \( \phi \), there is a \( \text{KL} \)-model \( M \) and a world \( w \in M \) such that \( M,w \models \phi \), and
Then KL has the polynomial-size model property, and KL-SAT ∈ NP.

Proof. Let \( \phi \) be a KL-satisfiable formula, and let \( M \) be a KL-model and \( w \in M \) meeting the prerequisites of the corollary. By Lemma 3.8, there is a submodel \( M' \) of \( M \) such that \( M', w \models \phi \), and

\[
\left| \left\{ v \in M' \mid \text{maxdepth}^{M'}(v) < c \right\} \right| \leq (c + 1) \cdot |\phi|^c.
\]

Therefore, since the conditions required in the prerequisites of the corollary are invariant under further restrictions of the model, assume without loss of generality that \( M \) already satisfies this condition. Now let \( M = T \cup C \cup B \), where

\[
T := \left\{ v \in M \mid \text{maxdepth}^M(v) < c \right\},
\]

\[
C := \left\{ v \in M \mid \text{maxdepth}^M(v) \geq c \text{ and } \text{maxheight}^M(v) \geq c \right\},
\]

\[
B := \left\{ w \in M \mid \text{maxheight}^M(w) < c \right\}.
\]

Note that \( T \) and \( B \) are not necessarily disjoint (\( T \) contains the nodes with only small depth at the “top” of the model, \( C \) represents the “center” of the model, and \( B \) is the “bottom.”). By the prerequisites of the corollary, we can assume that \( |C| \leq p(|\phi|) \), and by the above we know that \( |T| \leq (c + 1) \cdot |\phi|^c \). Now let \( A := \left\{ v \in M \mid \text{maxheight}^M(v) \geq c \right\} \).

It follows that \( A \subseteq T \cup C \), and hence \( |A| \leq |T| + |C| \leq (c + 1) \cdot |\phi|^c + p(|\phi|) \).

By Lemma 3.10, there is a submodel \( M' \) of \( M \) such that \( M', w \models \phi \) and \( |M'| \leq (|A| + 1) \cdot (1 + |\phi|)^c \), and hence by the above we have that \( |M'| \leq ((c + 1) \cdot |\phi|^c + p(|\phi|) + 1) \cdot (1 + |\phi|)^c \) (note that the cardinality of the size \( A \) defined with respect to the submodel \( M' \) is bounded by the cardinality of the original set \( A \)). Since \( p \) is a polynomial and \( c \) is a constant only depending on the logic KL, this is a polynomial size bound in \( |\phi| \), and therefore we have proven the polynomial-size model property. By Proposition 3.4, \( M' \) is a KL-model. The complexity result now follows from Proposition 2.2. \( \square \)

4 Universal Horn Formulas

We now consider a syntactically restricted case of universal first order formulas, namely Horn formulas. Many well-known logics can be expressed in this way.

4.1 Definitions

Usually, a Horn clause is defined as a disjunction of literals of which at most one is positive. If a positive literal occurs, then the clause can be written as an implication, since \( \overline{x_1} \lor \cdots \lor \overline{x_n} \lor y \) is equivalent to \( x_1 \land \cdots \land x_n \Rightarrow y \). If no positive literal occurs, then the clause \( (\overline{x_1} \lor \cdots \lor \overline{x_n}) \) can be written as \( x_1 \land \cdots \land x_n \Rightarrow \text{false} \). Since in the context of the frame language, an atomic proposition is of the form \( xRy \), the following is the natural version of Horn clauses for our purposes:
Definition 4.1. A universal Horn clause is a formula of the form \((x_1Rx_2) \land \cdots \land (x_{k-1}Rx_k) \implies (x_1Rx_j)\), or of the form \((x_1Rx_2) \land \cdots \land (x_{k-1}Rx_k) \implies \text{false}\), where all (not necessarily distinct) variables are implicitly universally quantified.

A universal Horn formula is a conjunction of universal Horn clauses. With universal Horn formulas, many of the usually considered graph properties can be expressed, like transitivity, symmetry, euclidicity, etc. In the following definition, we show how universal Horn clauses can be represented as graphs.

Definition 4.2. Let \(\hat{\varphi}\) be a universal Horn clause.

1. The prerequisite graph of \(\hat{\varphi}\), denoted with \(\text{prereq}(\hat{\varphi})\), consists of the variables appearing on the left-hand side of the implication \(\varphi\), where \((x_1,x_2)\) is an edge if the clause \((x_1Rx_2)\) appears.
2. If \(\hat{\varphi}\) is a universal Horn clause where the right-hand side of the implication in \(\hat{\varphi}\) is \(R(x,y)\), then the conclusion edge of \(\hat{\varphi}\), denoted with \(\text{conc}(\hat{\varphi})\), is the edge \((x,y)\). If the right-hand side of the implication is \(\text{false}\), then \(\text{conc}(\hat{\varphi})\) is the empty set.

4.2 Universal Horn Clauses and Homomorphisms

The definition of the prerequisite graph and the conclusion edge of a universal horn formula establishes a one-to-one correspondence between universal Horn clauses and their representation as graphs. These definitions allow us to relate truth of a Horn clause to homomorphic images of the involved graphs:

Proposition 4.3. 1. Let \(\hat{\varphi}\) be a universal Horn clause with \(\text{conc}(\hat{\varphi}) = (x,y)\). A graph \(G\) satisfies \(\hat{\varphi}\) if and only the following holds: For every homomorphism \(\alpha\): \(\text{prereq}(\hat{\varphi}) \cup \{x,y\} \rightarrow G\), \((\alpha(x),\alpha(y))\) is an edge in \(G\).
2. Let \(\hat{\varphi}\) be a universal Horn clause such that \(\text{conc}(\hat{\varphi}) = \emptyset\). Then a graph \(G\) satisfies \(\hat{\varphi}\) is and only if there is no homomorphism \(\alpha\): \(\text{prereq}(\hat{\varphi}) \rightarrow G\).

Proof. 1. Let \(\text{prereq}(\hat{\varphi}) = (x_1Rx_2) \land \cdots \land (x_{n-1}Rx_n)\), where all variables are implicitly universally quantified. First assume that \(G \models \hat{\varphi}\), and let \(\alpha\): \(\text{prereq}(\hat{\varphi}) \cup \{x,y\} \rightarrow G\) be a homomorphism. Due to the definition of \(\text{prereq}(\hat{\varphi})\), there are edges \((x_1,x_2),\ldots,(x_{n-1},x_n)\) in \(\text{prereq}(\hat{\varphi})\). Since \(\alpha\) is a homomorphism, this implies that \((\alpha(x_1),\alpha(x_2)),\ldots,(\alpha(x_{n-1}),\alpha(x_n))\) are edges in \(G\). Hence the nodes \(\alpha(x_1),\ldots,\alpha(x_n)\) satisfy the formula \(R(\alpha(x_1),\alpha(x_2)) \land \cdots \land R(\alpha(x_{n-1}),\alpha(x_n))\). Therefore, the nodes \(\{\alpha(v) \mid v \in \text{prereq}(\hat{\varphi}) \cup \{x,y\}\}\) satisfy the prerequisites of the clause \(\hat{\varphi}\). Since \(G \models \hat{\varphi}\), this implies that \((\alpha(x),\alpha(y))\) is an edge in \(G\).

Now for the other direction, assume that \(G\) fulfills the homomorphism property, and let \(\text{VAR}\hat{\varphi} = \{x_1,\ldots,x_n\}\). Let \(a_1,\ldots,a_n\) be nodes in \(G\) satisfying the prerequisite clause of \(\hat{\varphi}\), i.e., if \((x_{i_1},x_{i_2})\) is a clause in \(\text{prereq}(\hat{\varphi})\), then \((a_{i_1},a_{i_2})\) is an edge in \(G\). Then obviously the function \(\alpha\) mapping the variable \(x_i\) to the node \(a_i\), is a homomorphism from \(\text{prereq}(\hat{\varphi})\) to \(G\). By the prerequisites, we know that \((\alpha(x),\alpha(y))\) is an edge in \(G\). Hence, \(G\) satisfies the formula \(\hat{\varphi}\).

2. Analogous.

There is a natural correspondence between implications of these formulas and graph homomorphisms.
**Proposition 4.4.** 1. Let \( \hat{\varphi}_1 \) and \( \hat{\varphi}_2 \) be universal Horn clauses such that there exists a homomorphism \( \alpha: \text{prereq}(\hat{\varphi}_1) \rightarrow \text{prereq}(\hat{\varphi}_2) \), which maps the conclusion edge of \( \hat{\varphi}_1 \) to the conclusion edge of \( \hat{\varphi}_2 \). Then \( \hat{\varphi}_1 \) implies \( \hat{\varphi}_2 \).

2. Let \( \hat{\varphi}_1 \) and \( \hat{\varphi}_2 \) be universal Horn clauses such that \( \text{conc}(\hat{\varphi}_1) = \text{conc}(\hat{\varphi}_2) = \emptyset \), and let \( \alpha: \text{prereq}(\hat{\varphi}_1) \rightarrow \text{prereq}(\hat{\varphi}_2) \) be a homomorphism. Then \( \hat{\varphi}_1 \) implies \( \hat{\varphi}_2 \).

**Proof.** 1. Let \( \text{conc}(\hat{\varphi}_1) = (x, y) \), then by the prerequisites it follows that \( \text{conc}(\hat{\varphi}_2) = (\alpha(x), \alpha(y)) \). Now let \( G \) be a graph such that \( G \models \hat{\varphi}_1 \), and let \( \beta: (\text{prereq}(\hat{\varphi}_2) \cup \{\alpha(x), \alpha(y)\}) \rightarrow G \) be a homomorphism. By Proposition 4.3, it suffices to show that \((\beta(\alpha(x)), \beta(\alpha(y))) \) is an edge in \( G \). Since \( \alpha \) and \( \beta \) are homomorphisms, \( \beta \circ \alpha: \text{prereq}(\hat{\varphi}_1) \rightarrow G \) is a homomorphism as well. Since \( G \models \hat{\varphi}_1 \), Proposition 4.3 implies that \((\beta(\alpha(x)), \beta(\alpha(y))) \) is an edge in \( G \), as claimed.

2. Let \( G \) be a graph such that \( G \models \hat{\varphi}_1 \). Due to Proposition 4.3, to show that \( G \models \hat{\varphi}_2 \), it suffices that there is no homomorphism \( \beta: \text{prereq}(\hat{\varphi}_2) \rightarrow G \). Hence assume that such a homomorphism exists. Then \( \beta \circ \alpha \) is a homomorphism from \( \text{prereq}(\hat{\varphi}_1) \) into \( G \), which is a contradiction to Proposition 4.3, since \( G \models \hat{\varphi}_1 \).

\[ \square \]

### 4.3 Important Special Cases

We now consider special cases of Horn clauses, which will be central for the logics having satisfiability problems in NP. The following definition captures the case where the variables in the conclusion edge have a common predecessor in the prerequisite graph, but there is not necessarily a direct path between them. Using results about graphs satisfying generalizations of formulas of this type, we will be able to show all of the NP-containment results that the proof of the later classification theorem, Theorem 4.17, depends on.

**Definition 4.5.** Let \( \hat{\varphi}^{k-l} \) be the formula

\[
(wRx_1) \land (x_1Rx_2) \land \cdots \land (x_{k-1}Rx_k) \land (wRy_1) \land (y_1Ry_2) \land \cdots \land (y_{l-1}Ry_l) \implies (x_kRy_l),
\]

where all variables are universally quantified (and in the case that \( x_0 \) or \( y_0 \) appear in the formula, we replace them with \( w \)).

In Figure 1, we present the graph representation of the formula \( \hat{\varphi}^{2-4} \). The graph property described by these formulas is easy to see:

**Proposition 4.6.** Let \( G \) be a graph, and let \( k, l \in \mathbb{N} \). Then \( G \models \hat{\varphi}^{k-l} \) if and only if for any nodes \( w, x_k, y_l \in G \), if \( G \models w \rightarrow x_k \) and \( G \models w \rightarrow y_l \), then \((x_k, y_l)\) is an edge in \( G \).

This definition generalizes several well-known examples—in particular, a graph is reflexive if and only if it satisfies \( \hat{\varphi}^{0-0} \), symmetry is expressed with \( \hat{\varphi}^{1-0} \), and transitivity with \( \hat{\varphi}^{0-2} \). Finally, a graph is Euclidean iff it satisfies \( \hat{\varphi}^{1-1} \). Therefore, this notion allows us to capture many interesting graph properties, and it is not surprising that generalizations of this idea are the main ingredients for our polynomial size model proofs. We start with looking at some properties and implications of formulas of the form \( \hat{\varphi}^{k-l} \).

**Lemma 4.7.** Let \( 1 \leq k, l \in \mathbb{N} \), and let \( G \) be a graph such that \( G \models \hat{\varphi}^{k-l} \).
1. $G \models \phi^{l+k-1-l+k}$.
2. If $l = k + 1$, then for any $i \geq k$, $G \models \phi^{i-i+1}$.
3. There is some $k' \geq 1$ such that $G \models \phi^{i-i+1}$ for all $i \geq k'$.

Proof

1. Let $w$ be some node in $G$, such that $G \models w \rightarrow x_{l+k-1}$, and $G \models w \rightarrow y_{l+k}$, and let the (not necessarily distinct) intermediate vertices be denoted with $x_i, y_i$. Since $\hat{\phi}^{k-l}$ holds in $G$, this implies that there is an edge $(y_k,x_l)$. By choice of nodes, $G \models x_l \rightarrow x_{l+k-1}$. Combining these, we obtain a path of length $k$ from $y_k$ to $x_{l+k-1}$. On the other hand, $G \models y_k \rightarrow y_{k+l}$. Since $\hat{\phi}^{k-l}$ holds in $G$, it follows that there is an edge from $x_{l+k-1}$ to $y_{l+k}$, proving that $\hat{\phi}^{l+k-1-l+k}$ holds in $G$.

2. Clearly it suffices to show $G \models \hat{\phi}^{k+1-k+2}$, the claim for arbitrary $i$ follows inductively. Let $w, x_j, y_j$ be chosen such that $w = x_0 = y_0$, and there are edges $(x_j,x_{j+1})$ and $(y_j,y_{j+1})$. We need to show that there is an edge $(x_{k+1}, y_{k+2})$. Since $G \models \hat{\phi}^{k-k+1}$, it follows that $(y_k,x_{k+1})$ is an edge in $G$. Since there obviously is a path of length $k-1$ from $y_1$ to $y_k$, it follows that there is a path of length $k$ from $y_1$ to $x_{k+1}$. Since there also is a path of length $k+1$ from $y_1$ to $y_{k+2}$, it follows that there is an edge $(x_{k+1}, y_{k+2})$ in $G$, which concludes the proof.

3. This follows immediately from the above: from points 1 and 2, it follows that the claim holds for $k' := l + k - 1$.

The formula $\hat{\phi}^{k-l}$ is supposed to capture the case where the variables in the conclusion edge of a universal Horn clause have a common predecessor. But not all of these cases are covered with this formula. The above Figure 1 is a graphical representation of what the implication $\hat{\phi}^{2^{-1}}$ does. But what if this is only a subgraph of the prerequisite graph? In a more general case, the node $w$ and the nodes $x_k, y_l$ will have more predecessors and successors. Figure 2 gives an example of a more general formula. We will now see that this formula can be “simplified.” This simplification is not an equivalent transformation of the formula, but we construct a new formula which is implied by the original one. The one-sided implication suffices to show many of the results we need. The simpler formula is presented in Figure 3.

It is easy to see that every graph which satisfies the formula displayed in Figure 2 also satisfies the formula from Figure 3. This follows directly from Proposition 4.4, since the prerequisite graph from Figure 2 can obviously be mapped homomorphically to the prerequisite graph from Figure 3 (the homomorphism $\alpha$ is defined as $\alpha(t_1) := x_2, \alpha(s) := u_2, \alpha(u_3) := w$, and maps the other nodes to the ones with the same labels). Hence, if we can show NP-containment for all universal elementary modal logics extending the one defined by the later formula, this puts the logic defined by the original formula into NP as well.
Any universal Horn clause which can be mapped onto a tree can be embedded in a graph with certain properties, namely the properties of the formula we now define. Due to Corollary 2.7, it is natural that tree-like homomorphic images of our universal Horn formulas are of interest to us. These formulas capture the generalizations of $\bar{\varphi}^{k-l}$ mentioned above, where we demand that the nodes $w, x_k, y_l$ have a sufficient number of predecessors or successors. We again use the representation of Horn clauses as graphs.

**Definition 4.8.** For $k, l, p, q, r \in \mathbb{N}$, the formula $\bar{\varphi}_{w \geq p, x \geq q, y \geq r}^{k-l}$ is defined as the universal Horn clause displayed in Figure 4.

It should be noted that the notation $\bar{\varphi}_{w \geq p, x \geq q, y \geq r}^{k-l}$ suggests that $w, y, x$ can be compared to natural numbers, but what is meant in that notation is simply that the number of predecessors (successors, resp.) of $w, x_k, y_l$ can be compared to $p, q$, and $r$, respectively. Hence, if we use “natural names” for the vertices, i.e. we have vertices $w = x_0, x_1, \ldots$, and $w = y_0, y_1, \ldots$, then this ensures that the vertices up to $x_q$, $y_r$, and $w_p$ exist.

When proving that this formula holds in a graph, we will usually rely on the notation provided in Figure 4, and assume that there are nodes $w_p, \ldots, w_0 = w = x_0 = y_0, x_1, \ldots, x_q, x_1, \ldots, y_r$ with edges as seen in Figure 4, i.e., most of the time we do not mention the homomorphism explicitly.

The formula $\bar{\varphi}_{w \geq p, x \geq q, y \geq r}^{k-l}$ can be seen to be only a slight generalization of the formulas $\bar{\varphi}^{k-l}$ we already considered, as exhibited by the following proposition:

**Proposition 4.9.** Let $G$ be a graph, and let $k, l, p, q, r \in \mathbb{N}$. Then $G \models \bar{\varphi}_{w \geq p, x \geq q, y \geq r}^{k-l}$ if and only if the following condition holds: For any nodes $w, x_k, y_l \in G$, such that $w$ has a $p$-step predecessor, $x$ has a $q-k$-step successor and $y$ has an $r-l$-step successor, $G \models w \rightarrow^{k} x_k$ and $G \models w \rightarrow^{l} y_l$, it follows that $(x_k, y_l)$ is an edge in $G$.

The above proposition immediately implies the following:

**Proposition 4.10.** Let $G \models \bar{\varphi}_{w \geq p, x \geq q, y \geq r}^{k-l}$, and let $G'$ be the restriction of $G$ to the set

$$C := \{ w \in G \mid \text{maxdepth}^G(w) \geq p, \text{maxheight}^G(w) \geq \max(q-k, r-l) \},$$

then $G' \models \bar{\varphi}^{k-l}$.

This Proposition is one of the reasons why Corollary 3.11 is important: We can use it together with Proposition 4.10 to restrict our attention to the vertices in the “middle” of the graph, and then talk about the formula $\bar{\varphi}^{k-l}$ instead of $\bar{\varphi}_{w \geq p, x \geq q, y \geq r}^{k-l}$. This is the general approach for our NP-containment proofs, although some technical difficulties remain, as we will see in the proof of Theorem 4.13.

There obviously is a close relationship between formulas of the form $\bar{\varphi}_{w}^{k-l}$ and formulas of the form $\bar{\varphi}_{w \geq p, x \geq q, y \geq r}^{k-l}$. In particular, this relationship allows the “lifting” of implications, as is shown in the following easy lemma.
Lemma 4.11. Let \( p, q, r, k, l, k', l' \in \mathbb{N} \), and let \( \hat{\varphi}^{k-l} \) imply \( \hat{\varphi}^{k'-l'} \). Then \( \hat{\varphi}^{k-l} \) implies \( \hat{\varphi}^{k'-l'} \), where
\[
\begin{align*}
p' &= p, \\
q' &= k' + \max(q - k, r - l), \\
r' &= l' + \max(q - k, r - l).
\end{align*}
\]

Proof. Let \( G \) be a graph satisfying \( \hat{\varphi}^{k-l} \). Let \( w, x, y \) be the nodes in the graph connected as the nodes in the prerequisite graph of \( \hat{\varphi}^{k'-l'} \). Let \( G' \) be the graph \( G \) restricted to the set of vertices which have a \( p \)-step predecessor, and \( \max(q - k, r - l) \)-step successor in \( G \). Then, by choice of nodes, and Propositions 4.9 and 4.6, it follows that \( G' \models \hat{\varphi}^{k-l} \). Hence, due to the prerequisites, we know that \( G \models \hat{\varphi}^{k'-l'} \). In particular, since the nodes \( w, x, y \) satisfy the prerequisite graph of \( \hat{\varphi}^{k'-l'} \), we know that \( w, x, y \in G' \). Hence, it follows that \((x, y)\) is an edge in \( G' \), and therefore it is an edge in \( G \), as claimed. \( \square \)

We need three more implications between formulas of this kind for our later NP-results:

Lemma 4.12. Let \( p, q, r, k \in \mathbb{N}, k \geq 2 \), and let \( G \) be a graph.
\[
\begin{align*}
1 &. \text{If } G \models \hat{\varphi}^{k-0}, \text{ then } G \models \hat{\varphi}^{0-k^2}. \\
2 &. \text{If } G \models \hat{\varphi}^{k-0}_{\geq p, x \geq q, y \geq r}, \text{ then } G \models \hat{\varphi}^{k-0}_{\geq p', x \geq q', y \geq r'}, \text{ where}
&\quad p' = p, \\
&\quad q' = k - 1 + \max(q - k, r), \\
&\quad r' = k^3 + \max(q - k, r).
\end{align*}
\]
\[
3. \text{Let } k \in \mathbb{N}. \text{ Then } \hat{\varphi}^{0-k} \text{ implies } \hat{\varphi}^{0-k^2}.
\]

Proof. Let \( w_p, \ldots, w = x_0, \ldots, w = y_0, \ldots \) be nodes in the graph fulfilling the prerequisite graph of the respective formulas.
\[
\begin{align*}
1 &. \text{By the } \hat{\varphi}^{k-0}-\text{property, it is clear that } (y_k, y_0), (y_{2k}, y_k), \ldots, (y_{k^2}, y_{(k-1)k}) \text{ are edges in } G. \text{ Hence, } G \models y_k \xrightarrow{k} w. \text{ Again due to the property, it follows that } (w, y_{k^2}) \text{ is an edge in } G, \text{ as required.}
\\
2 &. \text{Consider the subgraph } G' := G \cap \{w, x_1, \ldots, x_{k-1}, y_1, \ldots, y_{k^2}\}. \text{ By choice of } p', q', r', \text{ every node in } G' \text{ has a } p \text{-step predecessor, a } q - k \text{-step successor, and an } r \text{-step successor in } G. \text{ Hence, } G' \models \hat{\varphi}^{k-0}. \text{ Due to part 1, this implies that } G' \models \hat{\varphi}^{0-k^2}. \text{ Hence, in } G' \text{ there are edges } (w, y_{k^2}), (y_{2k}, y_{2k^2}), \ldots, (y_{(k-1)k^2}, y_{k^3}). \text{ Therefore, due to the } \hat{\varphi}^{k-0}-\text{property, it follows that } (y_k, w) \text{ is an edge in } G'. \text{ Hence, it follows that } G \models y_k \xrightarrow{k} x_{k-1}. \text{ The } \hat{\varphi}^{k-0}-\text{property implies that } (x_{k-1}, y_{k^3}) \text{ is an edge, as required.}
\\
3 &. \text{This follows from Lemma 4.20, since } k\text{-transitivity implies } k+1 \cdot (k-1) = k^2\text{-transitivity.}\)
\]

4.4 NP upper complexity bounds

In this section, we prove NP-containment results for logics defined by universal Horn clauses. We show that “most” of the logics of the form \( K(\hat{\varphi}^{k-l}_{\geq p, x \geq q, y \geq r}) \) give rise to a satisfiability problem in NP.

It is comparably easy to show that a logic of the form \( K(\hat{\varphi}^{k-l+1}) \) leads to a satisfiability problem in NP, by carefully “copying” vertices and adding the correct neighbors. However,
while this process leads to a model which still satisfies $\hat{\varphi}^{k-k+1}$, it does not give the desired result that the problem can be solved in NP for all universal Horn logics which are extensions of $K(\hat{\varphi}^{k-k+1})$. In order to prove this, our model-manipulations must be consistent with the conditions of Theorem 3.3. In the proof of the following theorem, we construct a small model using only restriction. Note that the proof for this theorem is the only occasion where we actually prove the polynomial-size model property by explicitly constructing the model. All further NP-results make use of Theorem 4.13, by showing that the logic in question is an extension of one of the logics that this theorem deals with. For example, the logic $K4B$ satisfies the conditions of this Theorem. It is therefore a central theorem for our complexity classification.

The proof relies on one later result, namely the second case of Theorem 4.33. However, the proof for this case of Theorem 4.33 does not rely on any facts about the classification algorithm studied later. In the following, a graph $G$ with root $w$ is $w$-canonical if for every node $x \in G$, if $G \models w \xrightarrow{i} x$ and $G \models w \xrightarrow{j} x$, then $i = j$. Note that this does not mean that every node $x$ is reachable on only one path from the root, but that all paths from the root $w$ to $x$ have the same length. We say that a graph $G$ is canonical if it is $w$-canonical for a root $w$ of $G$. It is easy to see that a graph is canonical if and only if it can be homomorphically mapped onto a strict line.

**Theorem 4.13.** Let $\hat{\varphi}$ be a universal Horn formula implying $\hat{\varphi}^{k-k+1}$ for some $k, p, q, r \in \mathbb{N}, k \geq 1$. Then $K(\hat{\varphi})$ has the polynomial-size model property, and $K(\hat{\varphi})$-SAT $\in$ NP.

**Proof.** Due to Proposition 2.2, it suffices to prove the polynomial-size model property for $K(\hat{\varphi})$. Hence, let $\varphi$ be a $K(\hat{\varphi})$-satisfiable modal formula, and, following Proposition 3.7, let $M$ be a $K(\hat{\varphi})$-model with root $w$ such that $M, w \models \varphi$. Define $maxline$ to be the maximal length of a strict line on which $\varphi$ is satisfied, and $maxline$ to be zero in the case that $\varphi$ is satisfied on every strict line (note that due to Proposition 3.4, if $\varphi$ is satisfied on the strict line of length $i$, then it is also satisfied on the strict line of length $i-1$). This number is obviously a constant depending only on $\varphi$. If $\varphi$ is satisfied on every strict line, we can, due to Theorem 4.33, assume that $M$ is $w$-canonical. Note that we will work with restrictions of the model during the course of the proof—since any restriction of a canonical graph having the same root is still canonical, the submodels we consider later are canonical as well. Now, define

$$A := \left\{ v \in M \mid \text{maxdepth}^M(v) \leq p + k + 2 \right\} \cup \{w\},$$
$$S := \left\{ v \in M \mid \text{maxheight}^M(v) \geq \max(q - k, r - k - 1) \right\},$$
$$C := (M \setminus A) \cap S.$$  

The set $C$ contains all the nodes in $M$ which have “enough” predecessors and successors to ensure that the formula $\hat{\varphi}_{w \geq p, x \geq q, y \geq r}^{k-k+1}$ gives us all the necessary edges that we are interested in. To be more precise, we show the following fact:

**Fact 1** For all $i \geq 0$, it holds that $C \models \hat{\varphi}^{i-i+1}$.

**Proof.** It suffices to show the claim for $i = 1$. The result for arbitrary $i$ then follows from Lemma 4.7, and the observation that $\hat{\varphi}^{0-1}$ is true in any graph. Hence, we show $C \models \hat{\varphi}^{1-2}$: Let $w', x_1, y_1, y_2$ be vertices from $C$, such that they satisfy the prerequisites of $\hat{\varphi}^{1-2}$, i.e., let $(w', x_1), (w', y_1),$ and $(y_1, y_2)$ be edges in $M$. Since $w' \in C$, we know that $w'$ has a $k - 1$-step predecessor $w''$, which in turn has a $p$-step predecessor. Due to the edges mentioned
above, it obviously holds that $M \models w \xrightarrow{k} x_1$, and $M \models w \xrightarrow{k+1} y_2$. Since both $x_1$ and $y_2$ are elements of $C$, they have the required number of successors, and therefore, since $M \models \phi_{w,x\geq p,y\geq r}^{{k-k+1}}$, Proposition 4.9 implies that $(x_1,y_2)$ is an edge in $M$, as claimed. \hfill \Box

Due to Lemma 3.8, we can assume, without loss of generality, that $|A|$ is polynomially bounded (in the length of $\phi$). We can additionally assume, again due to Lemma 3.8, that the number of nodes $v$ with maxdepth$^M(v) \leq p+k+3$ is also restricted by a polynomial. Due to Proposition 3.7, we can also assume that every world in $M$ can be reached from $w$ in $md(\phi)$ steps. Therefore, we can choose a set $G \subseteq M$ with the following properties:

- $G$ is polynomially bounded,
- $G \subseteq C$,
- for every node $u \in A$ and every $v \in C$ such that $(u,v)$ is an edge in $M$, and every subformula $\psi$ of $\phi$ such that $M,v \models \psi$, there is a node $v' \in G$ such that $(u,v')$ is an edge in $M$, and $M,v' \models \psi$,
- every node in $C$ can be reached on a path from some node in $G$. Additionally, if $\phi$ is satisfied on every strict line, every node in $C$ can be reached from some node in $G$ on a path of length at most $md(\phi)$.

Such a set can be chosen with an application of the technique used to prove Lemma 3.8:

**Fact 2** A set $G$ can be chosen with the properties above.

**Proof.** First consider the case that $\hat{\phi}$ is satisfied on every strict line. In this case, since $M$ is $w$-canonical, we know that $A$ is exactly the set of vertices on the first $p+k+2$ levels of $M$ (the $i$-th level is the set of vertices $v$ in $M$ such that $M \models w \overset{i}{\rightarrow} v$. Note that a canonical graph can only have one root). Since $M$ is $w$-canonical, we can simply choose $G$ to be the set of those nodes at level $p+k+3$ in the model $M$ which are also elements of $C$. Since in a canonical model $M$, for each node $v$ maxdepth$^M(v)$ is exactly the level of $v$ in $M$, this is a set of polynomial size, and every path from $w$ to a node in $C$ passes through $G$. Further, every successor $v \in C$ of a node $u \in A$ is trivially a member of $G$. Since every node in $M$ can be reached from the root $w$ in at most $md(\phi)$ steps, the claim follows.

Now consider the case where $\hat{\phi}$ is not satisfied on every strict line. For each world $u \in M$, let $F_u := \{ \psi \in sf(\phi) \mid M,u \models \hat{\psi} \}$, and let $W_u$ be a subset of successors of $u$ in $M$ such that for every $\psi \in F_u$, there is a world $v \in W_u$ such that $(u,v)$ is an edge in $M$, $M,v \models \psi$, and $|W_u| \leq |sf(\phi)|$. If it is possible to choose $v \in C$, then do so.

Define $M_0 := \{w\}$, and for each $i \in \mathbb{N}$, let $M_{i+1} := \bigcup_{u \in M_i} W_u \setminus \bigcup_{j=0}^{i-1} M_j$. Now define $M'$ to be the restriction of $M$ to the worlds in $\bigcup_{i \in \mathbb{N}} M_i$. We show that $M'$, $w \models \phi$ by proving that $M'$ satisfies the conditions of Lemma 3.5. By construction, we know that $w \in M'$. Now let $u \in M'$, and let $\psi$ be a subformula of $\phi$ with $M,u \models \hat{\psi}$. Then $\phi \in F_u$, and therefore there is a world $v \in W_u$ with $M,u \models \phi$. Since $u \in M'$, there is some $i$ with $u \in M_i$. It follows that $W_u \subseteq \bigcup_{j=0}^{i+1} M_j \subseteq M'$, and hence $v \in M'$ as claimed. Since $M'$ is a restriction of $M$, we know that $M'$ also is a $K(\hat{\phi})$-model, and inherits all size restrictions on submodels that we have already established. Therefore, we can, without loss of generality, assume that $M' = M$.

Now define $G := \bigcup_{i=0}^{p+k+3} M_i \cap C$. By the proof of Lemma 3.8, since $p+k+3$ is a constant, we conclude that $G$ is polynomial in $|\phi|$. By construction, $G \subseteq C$. We prove that every node $u \in C$ can be reached on a path from a node in $C$. For every $u \in M$, there exists a sequence of nodes $w = u_0,u_1,\ldots,u_n = u$ such that for all relevant $i$, $u_{i+1} \in W_{u_i}$, $(u_i,u_{i+1})$ is an edge in $M$, and $u_i \in M_i \setminus \bigcup_{j=0}^{i-1} M_i$. Let $t$ be minimal such that $u_t \in C$. Since $w \notin C$, it follows that $t \geq 1$, and we know that such a $t$ exists, since $u_n = u \in C$. It suffices to prove that
FACT 3 Let \( g \in C \). Then \( M^C_g \) is a graph with root \( g \) such that \( M^C_g \models \varphi^{i-i+1} \) for all \( i \geq 0 \).

For any \( g \in C \) and \( i \in \mathbb{N} \), we define
\[
L^g_i := \{ v \in M^C_g \mid M \models g \xrightarrow{i} v \}.
\]

We observe the following fact:

FACT 4 Let \( g \in C \), \( i \in \mathbb{N} \), \( x \in L^g_i \), and \( y \in L^g_{i+1} \). Then \((x, y)\) is an edge in \( M \).

Proof. This immediately follows from Fact 3 and the definition of \( L^g_i \) : By definition, it follows that \( C \models g \xrightarrow{i} x \), and \( C \models g \xrightarrow{i+1} y \). From Fact 1, we know that \( C \models \varphi^{i-i+1} \), and hence there is an edge \((x, y)\) in \( C \) as claimed.

Graphs fulfilling the formula \( \varphi^{i-i-2} \) are “layered.” Whenever there are nodes \( x \) and \( y \) such that there is a path of length \( i \) from the root to \( i \), and of length \( i+1 \) to the node \( y \), then there is an edge between \( x \) and \( y \). From the definition, it is obvious that for any subgraph \( X \) of \( C \) with root \( g \), that \( X \) is canonical, if for every pair of natural numbers \( i \neq j \), it follows that \( L^g_i \cap L^g_j \cap X = \emptyset \). We make a distinction between those elements in \( G \) which lead to a canonical graph, and those which do not. In light of Lemma 3.6, it is natural that we do not need to look at the entire graph, but can ignore nodes which do not have “short” paths from the root of the graph leading to it. For a natural number \( b \), we say that \( X \) is \( b \)-canonical, if for all \( 0 \leq i < j \leq b \) it holds that \( L^g_i \cap L^g_j \cap X = \emptyset \).

We define the following:
\[
G_{\text{can}} := \{ g \in G \mid M^C_g \text{ is md } (\phi) \text{-canonical} \},
\]
\[
G_{\text{non-can}} := \{ g \in G \mid M^C_g \text{ is not md } (\phi) \text{-canonical} \}.
\]

It is obvious that \( G = G_{\text{can}} + G_{\text{non-can}} \). For each \( g \in G_{\text{non-can}} \), let \( i(g), j(g) \) denote natural numbers, and let \( n(g) \) denote some node such that \( n(g) \in L^g_{i(g)} \cap L^g_{j(g)} \), with \( 0 \leq i(g) < j(g) \leq \text{md } (\phi) \), and \( i(g) \) is minimal with these properties. In particular, observe that \( i(g) \) and \( j(g) \) are polynomial in \( |\phi| \).

We now show that for \( g \in G_{\text{non-can}} \), the graph \( M^C_g \setminus M^C_{n(g)} \) still has root \( g \), unless it is empty. It particular, this means that it makes sense to ask if these graphs are \( \text{md } (\phi) \)-canonical.
Fact 5 Let $g \in G_{\text{non-can}}$, such that $M^C_g \setminus M^C_{n(g)}$ is not empty. Then $M^C_g \setminus M^C_{n(g)}$ has root $g$.

Proof. Assume that this is not the case, i.e., that there is some $v \in M^C_g \setminus M^C_{n(g)}$, and in this graph, there is no path from $g$ to $v$. Since $v$ is an element of $M^C_g$, a path from $g$ to $v$ exists in the original model $M$. Let $g \rightarrow v_1 \rightarrow \cdots \rightarrow v_i \rightarrow v$ be this path. Since the path does not exist in the graph $M^C_g \setminus M^C_{n(g)}$, it follows that one of the $v_j$ must be an element of $M^C_{n(g)}$. Since there is a path from this $v_j$ to $v$, it follows that $v$ is an element of $M^C_{n(g)}$ as well, which is a contradiction. 

The nodes $n(g)$ can be seen as “minimal non-canonical points” in $M^C_g$. The following makes this more precise:

Fact 6 Let $g \in G_{\text{non-can}}$. Then $M^C_g \setminus M^C_{n(g)}$ is md ($\phi$)-canonical.

Proof. Due to Fact 5, we know that $M^C_g \setminus M^C_{n(g)}$ has root $g$, and by definition this graph is a subset of $C$. Assume that it is not md ($\phi$)-canonical. Then there exist natural numbers $i_1 < i_2 \leq \text{md} (\phi)$, and some node $v \in (L^g \cap L^g_{i_2}) \setminus M^C_{n(g)}$. Due to the minimality of $i(g)$, it follows that $i(g) \leq i_1$, and hence $i(g) < i_2$. In particular, it follows by induction on Fact 4 that there is a path from $n(g) \in L^g_{i_1}$ to $v \in L^g_{i_2}$, i.e., $v \in M^C_{n(g)}$. This is a contradiction. 

As our next connectivity result, we show the following:

Fact 7 Let $g \in G_{\text{non-can}}$, and let $x \in L^g_{i_1}$, $y \in L^g_{i_2}$, where $i_1$ and $i_2$ are natural numbers such that $i_2 \equiv i_1 + 1 \mod (j(g) - i(g))$. Then $(x, y)$ is an edge in $M$.

Proof. Since $n(g) \in L^g_{i_1} \cap L^g_{j(g)}$, and $j(g) > i(g)$, by induction on Fact 4 we know that $M \models n(g)^{\equiv j(g) - i(g)}_D n(g)$, and obviously this implies that for every multiple of $j(g) - i(g)$, a path of that length exists in $M$ from $n(g)$ to $n(g)$ itself. Now assume that $i_2 = i_1 + j(j(g) - i(g))$ for some integer $j$. We make a case distinction:

Case 1: $j \geq 0$ The above implies that $M \models n(g)^{i_1 + j(j(g) - i(g))}_D x$, and by choice of $j$, we also know that $M \models n(g)^{i_1 + j(j(g) - i(g)) + 1}_D y$. Since all of the involved nodes are elements of $C$, and from Fact 1 we know that $C \models \phi^{\equiv l + 1}_D$ for all $l \geq 0$, this implies that there is an edge from $x$ to $y$ in $M$, as claimed.

Case 2: $j < 0$ By choice of $i_1$, we know that $M \models n(g)^{i_1}_D x$. By choice of $j$, the above and since $-j$ is positive, we also know that $M \models n(g)^{i_1 - j(j(g) - i(g)) + 1}_D y$. Hence the existence of the edge $(x, y)$ again follows from Fact 1.

We therefore have the following structure of the model $M$: By definition, $M = A \cup C \cup M \setminus S$, and by choice of $G$, every node in $C$ can be reached on a path from a node in $G$. Therefore, $C = \bigcup_{g \in G} M^C_g$, and hence $M$ can be written as the union of sub-graphs as follows:

$$M = A \cup \bigcup_{g \in G_{\text{can}}} M^C_g \setminus M^C_{n(g)} \cup \bigcup_{g \in G_{\text{non-can}}} (M^C_g \setminus M^C_{n(g)}) \cup \bigcup_{g \in G_{\text{non-can}}} M^C_{n(g)} \cup (M \setminus S).$$

Recall that the set $G$ is polynomial in $|\phi|$. Due to Corollary 3.11, it suffices to restrict the “middle part” of this equation, i.e., the components except $A$ and $M \setminus S$, to polynomial
size in order to obtain the desired polynomial model. In order to do this, we will now prove further connectivity results for the sub-models $M^C_n(g)$. It is important to note that due to Proposition 3.4, all of these submodels inherit all of the properties of their respective super-models which can be expressed by a universal first order formula.

The idea behind the construction is the following: For each node in the original model, we add enough successors in our new model to ensure that the new model satisfies the conditions of Lemma 3.6. For the $\text{md}(\phi)$-canonical submodels, we know that we can stop adding nodes at depth $\text{md}(\phi)$, since other nodes cannot be reached on a shorter path from the root. For the non-$\text{md}(\phi)$-canonical submodels, we cannot do this, but we also do not need to: Due to Fact 7, we have “circular” edges, hence we know that nodes in a “low” level also are in a “high” level.

For the construction, let $G_{\text{can}} = \{g_1, \ldots, g_n\}$, and let $G_{\text{non-can}} = \{g_{n+1}, \ldots, g_{n+m}\}$. For $1 \leq i \leq m$, define $g_{n+i} := n(g_{n+i})$ (note that in this case, by definition $g_{n+i}$ is a member of $G_{\text{non-can}}$). For $1 \leq i \leq n + 2m$, define

$$N_i := \begin{cases} M^C_{g_i}, & \text{if } 1 \leq i \leq n, \\ M^C_{g_i} \setminus M^C_{n(g_i)}, & \text{if } n + 1 \leq i \leq n + m, \\ M^C_{g_i}, & \text{if } n + m + 1 \leq i \leq n + 2m. \end{cases}$$

Note that by definition and Fact 5 it follows that $N_i$ is a graph which is either empty or has root $g_i$. It also follows from the above that for $1 \leq i \leq n + m$, the graph $N_i$ is $\text{md}(\phi)$-canonical. Since the union over all $N_i$ is the same as the union over all $M^C_g$ (for all $g \in G$), it follows that

$$M = A \cup \left( \bigcup_{1 \leq i \leq n + 2m} N_i \right) \cup (M \setminus S).$$

Note that $n + 2m$ is polynomial in $|\phi|$, since $G$ is. We need one helpful fact about the non-$\text{md}(\phi)$-canonical $N_i$-models:

**Fact 8** Let $i \in \{n + m + 1, \ldots, n + 2m\}$, i.e., let $N_i$ be not $\text{md}(\phi)$-canonical. Then $N_i = \bigcup_{j=0}^{j(g_i-m)-i(g_i-m)} L^C_j g_i$.

**Proof.** The inclusion $\supseteq$ is obvious, since $N_i = M^C_{g_i}$. For the inclusion $\subseteq$, let $u \in N_i$. Since, by definition, $g_i = n(g_i-m)$, and hence $N_i = M^C_{n(g_i-m)}$, there is a minimal $j \in \mathbb{N}$ such that $u \in L^C_j (g_i-m)$. If $j \leq j(g_i) - i(g_i)$, then the claim holds. Therefore, assume that $j > j(g_i-m) - i(g_i)$, and thus $j' := j - (j(g_i-m) - i(g_i-m)) > 0$. Since $L^C_{j(g_i-m)} \neq \emptyset$, we also know that $L^C_{j' + 1} \neq \emptyset$, therefore let $v \in L^C_{j' + 1} (g_i-m)$. By definition, it holds that $j = (j' + 1) + (j(g_i-m) - i(g_i-m))$, and in particular, $j \equiv (j' + 1) \mod (j(g_i-m) - i(g_i-m))$. By Fact 7, we know that there is an edge from $v \in L^C_{j' + 1} (g_i-m)$ to $u \in L^C_j (g_i-m)$. This implies, by definition, that $u \in L^C_j (g_i-m)$, and by minimality of $j$, we know that $j \leq j'$, and hence $j(g_i-m) - i(g_i-m) \leq 0$, i.e., $j(g_i-m) \leq i(g_i-m)$, a contradiction. \hfill \Box

We now define a series of models $M_i$ for $0 \leq i \leq n + 2m$, which, step by step, integrate “enough” vertices of the original model $M$ to ensure that the formula $\phi$ still holds, but restrict the size to a polynomial. As the induction start, we define $M_0 := A$. For $i \geq 1$, the construction is as follows:

- Add every world from $M_{i-1}$ to $M_i$. 

add \( g_i \) to \( M_i \).

- If \( 1 \leq i \leq m + n \), i.e., if \( N_i \) is \( \text{md} (\phi) \)-canonical, then for each \( 0 \leq j \leq \text{md} (\phi) + \text{maxline} + 1 \), and each formula \( \psi \in \mathsf{sf} (\phi) \) : if there is a world \( v \in L^g_{j_i} \) such that \( M, v \models \psi \), then add one of these worlds into \( M_i \).

- If \( n + m + 1 \leq i \leq n + 2m \), i.e., if \( N_i \) is not \( \text{md} (\phi) \)-canonical, then for each \( 0 \leq j \leq j(g) - i(g) + 1 \), and each formula \( \psi \in \mathsf{sf} (\phi) \), perform the following:
  - If there is a world \( v \in L^g_{j_i} \) such that \( M, v \models \psi \), then add one of these worlds \( v \) into \( M_i \).
  - If there is a world \( u \) which has been added into one of the \( \text{md} (\phi) \)-canonical submodels in the step above, and there is a successor \( v \in L^g_{j_i} \) of \( u \) such that \( M, v \models \phi \), then add one of these worlds \( v \) into \( M_i \).

By construction, since \( j(g) \) and \( i(g) \) are polynomial in \( |\phi| \) for each \( g \in G_{\text{non-can}} \), and \( |\mathsf{sf} (\phi)| \leq |\phi| \), there are only polynomially many worlds in \( M_{n+2m} \). We now define \( M' \) to be the model \( M_{n+2m} \cup (M \setminus S) \). Then the set of worlds in \( M' \) which have \( \max (q - k, r - k - 1) \)-step successor and a \( p + k + 3 \)-step predecessor is polynomially bounded, since this is a subset of \( M_{n+2m} \). Hence, due to Corollary 3.11, it suffices to show that \( M', u \models \phi \) in order to exhibit a model of \( \phi \) which is polynomial in size. In order to prove this, we show that \( M' \) satisfies the conditions of Lemma 3.6. Since \( v \in A \) by definition and \( M_0 = A, w \) is an element of \( M' \) by definition. Therefore, let \( u \) be an element of \( M' \), and let \( \psi \) be a subformula of \( \phi \), such that \( M, u \models \psi \), and let \( a \) be a natural number such that \( M \models w \rightarrow u, \) and \( 1 + a + \text{md} (\psi) \leq \text{md} (\phi) \). It suffices to show that there is a world \( v \in M' \) such that \( M, v \models \psi \), and \( (u, v) \) is an edge in \( M \).

Since \( M, u \models \psi \), there is a world \( v' \in M \), such that \( (u, v') \) is an edge in \( M \), and \( M, v' \models \psi \).

There are several cases to consider. If \( v' \in A \), then by the construction of \( M' \), \( v' \) is an element of \( M' \) as well. Hence we can choose \( v \) to be \( v' \). If \( u \in A \), and \( v' \notin A \), then it holds that either \( v' \in C \) or \( v' \notin S \). If \( v' \notin S \), then we know that \( v' \in M' \), and we can choose \( v = v' \). Hence assume that \( v' \in C \). In this case, by choice of \( G \), there is a world \( v'' \in G \), such that \( M, v'' \models \psi \), and \( (u, v'') \) is an edge in \( M \). Since \( G \subseteq M' \), we can choose \( v = v'' \). If \( u \in M \setminus S \), then \( v' \in M \setminus S \) holds as well.

Therefore, it remains to consider the case \( u, v' \in C \). First, assume that there is no \( i \in \{1, \ldots, n + 2m\} \) such that \( u, v' \in N_i \). Since \( u \in C \), it follows that \( u \in N_i \) for some \( i \).

Since \( v' \notin N_i \), \( N_i \) must be \( \text{md} (\phi) \)-canonical, since all other \( N_i \) contain all successors from \( C \) to nodes in \( N_i \). It follows that \( u \in M^C_{g_i} \), and therefore \( v \in M^C_{g_i} \) holds as well. Since \( v' \notin N_i \), there is a non-\( \text{md} (\phi) \)-canonical submodel \( N_j \) such that \( v' \in N_j = M^C_{n(g_j - m)} \), and hence, due to Fact 8, there is some \( j' \in \{0, \ldots, j(n(g_j - m)) - i(n(g_j - m)) \} \) such that \( v' \in L^g_{j'} \). In this case, a node \( v \) fulfilling the requirements has been added to the model \( M' \) due to the last condition in the construction. Now assume that there is some \( i \in \{1, \ldots, n + 2m\} \), such that \( u, v' \in N_i \). If it is possible to choose this \( i \) in such a way that \( N_i \) is not \( \text{md} (\phi) \)-canonical, then we do so.

In particular, since \( (u, v') \) is an edge in \( M \), there is some natural number \( j \) such that \( u \in L^g_{j_i} \), and \( v' \in L^g_{j_i+1} \). If it is only possible to choose \( i \) leading to a \( \text{md} (\phi) \)-canonical model \( N_i \), then choose \( i \) and \( j \) in such a way that \( j \) is minimal with this property. We make a case distinction.

**Case 1:** \( i \in \{n + m + 1, \ldots, n + 2m\} \), i.e., \( N_i \) is not canonical. By Fact 8, due to the minimality of \( j \), we know that \( j \leq j(g_i - m) - i(g_i - m) \). Now, since \( u \in L^g_{j_i} \) and \( v' \in L^g_{j_i+1} \), it follows that \( j + 1 \leq j(g_i - m) - i(g_i - m) + 1 \). Hence, by construction there is a \( v \in L^g_{j_i+1} \cap M' \), such that \( M, v \models \psi \), and due to Fact 4, there is an edge \( (u, v) \) in \( M \).
Case 2: $i \in \{1, \ldots, n + m\}$, i.e., $N_i$ is $\text{md}(\phi)$-canonical. If $j + 1 \leq \text{md}(\phi) + \text{maxline} + 1$, then, due to the construction of $M'$, a world $v$ from $L^g_{i+1}$ satisfying $\psi$ was added in the construction, and due to Fact 4, $(u, v)$ is an edge in $M$.

Therefore, assume that $j + 1 > \text{md}(\phi) + \text{maxline} + 1$, i.e., $j \geq \text{md}(\phi) + \text{maxline} + 1$. Since $N_i$ is $\text{md}(\phi)$-canonical, we know that $u$ cannot be reached from $q$, with a path shorter than $\text{md}(\phi)$ steps. Due to the choice of $i$ and $j$, we also know that $u$ does not appear in a non-$\text{md}(\phi)$-canonical submodel (otherwise, since the non-$\text{md}(\phi)$-canonical submodels are successor-closed there would be a non-$\text{md}(\phi)$-canonical submodel $N_i$ containing both $u$ and $v'$, and we would have chosen this), and in each $\text{md}(\phi)$-canonical submodel where $u$ appears, it has depth of at least $\text{md}(\phi) + 1$.

If $\hat{\phi}$ is satisfied on every strict line, we know that every node from $C$ can be reached from an element in $G$ with at most $\text{md}(\phi)$ steps, which is a contradiction.

Now consider the case that $\hat{\phi}$ is not satisfied on every strict line. Since in the $\text{md}(\phi)$-canonical submodels, strict lines having the length of the depth of the submodel appear, and $\hat{\phi}$ is satisfied in $M$, we know that these submodels cannot have depth of more than $\text{maxline}$. Therefore, we know that $j + 1 \leq \text{maxline}$, a contradiction.

Hence, we know that $M', w \models \phi$, concluding the proof of Theorem 4.13. □

We will now show that Theorem 4.13 implies NP-results for a number of related logics. The following theorem shows that modal logics for classes of frames which fulfill a natural generalization of the Euclidean property have the polynomial-size model property, and hence can be solved in NP. Note that the case $k = l = 1$ of Corollary 4.14 follows from the main result of [HR07]. Our results and theirs are incomparable: They achieve the NP-result for all normal modal logics extending what in our notation is $K(\bar{\phi}^{1-1})$, where our results only hold for logics defined by universal Horn clauses (although the proof of Theorem 4.13 in most cases gives the NP-result for all extensions of the logic which are defined by universal formulas over the frame language—the only exception is the case where the graph formula is satisfied on every strict line, note that this case needed special treatment in the proof of Theorem 4.13, and relies on the tree-like property for these logics proven later in Theorem 4.33). We achieve NP-results for many logics which are not extensions of $K(\bar{\phi}^{1-1})$, but do not prove these results for all extensions of these logics.

With the preceding theorem and the results on the implication for formulas of the form $\bar{\phi}^{k-1}_{p \geq p', x \geq q, y \geq r}$, we obtain the following corollary:

**Corollary 4.14.** Let $\bar{\phi}$ be a universal Horn formula implying $\bar{\phi}^{k-1}_{p \geq p', x \geq q, y \geq r}$ for some $k, l, p, q, r \in \mathbb{N}$, such that one of the following conditions holds:

- $1 \leq k, l$.
- $l = 0$ and $k \geq 2$.

Then $K(\bar{\phi})$ has the polynomial-size model property, and $K(\bar{\phi}) \cdot \text{SAT} \in \text{NP}$.

**Proof.** First assume that $1 \leq k, l$. Lemma 4.7 shows that $\bar{\phi}^{k-1}$ implies $\bar{\phi}^{k'-k+1}_{p' \geq p, x \geq q, y \geq r}$ for some $k' \geq 1$. From Lemma 4.11, we conclude that $\bar{\phi}^{k-1}_{p \geq p', x \geq q, y \geq r}$ implies $\bar{\phi}^{k'-k+1}_{p' \geq p, x \geq q, y \geq r}$ for some natural numbers $p', q'$, and $r'$. Therefore, $\bar{\phi}$ also implies $\bar{\phi}^{k'-k+1}_{p' \geq p, x \geq q, y \geq r'}$, and due to Theorem 4.13, $K(\bar{\phi})$ has the polynomial-size model property and $K(\bar{\phi}) \cdot \text{SAT} \in \text{NP}$.

For the case $l = 0$ and $k \geq 2$, observe that due to Lemma 4.12, $K(\bar{\phi})$ is also an extension of $K(\bar{\phi}^{k-1}_{p \geq p', x \geq q, y \geq r})$ for some constants $p', q', r'$. Due to the prerequisites, it holds that $1 \leq k - 1 \leq k^2 - 2$. Hence, the result follows from the case above. □
Corollary 4.14 covers all cases obtained from formulas of the form $\varphi_{w\geq p, x\geq q, y\geq r}^{k-1}$ where we possibly could expect the polynomial size property to hold: The requirement demanding that $k \geq 2$ is crucial, since the formula $\varphi_{w\geq 0, x\geq 0, y\geq 0}^{1-0}$ is satisfied in any symmetric graph, and hence satisfiability problems for the corresponding logics are PSPACE-hard due to Corollary 2.5. Additionally, logics defined by formulas of the form $\varphi_{w\geq 0, x\geq 0, y\geq 0}^{0-k}$ are PSPACE-hard as well: This formula is satisfied in every reflexive, transitive graph, and therefore the complexity result follows from Theorem 2.8. We therefore have proven that for Horn formulas defined by these single clauses, as soon as they are not satisfied in various combinations of strict, transitive, reflexive, and symmetric trees, they already imply the polynomial-size model property.

Up to now, we only considered the effect of a single universal Horn clause. However, there are many cases where the logics defined by individual formulas have a PSPACE-hard satisfiability problem, but the complexity of the problem for logic defined by their conjunction drops to NP. A well-known example for such a case is the modal logic $K4B$, which is the logic over graphs which are both symmetric and transitive. On their own, both of these properties lead to PSPACE-hard logics, but their conjunction gives a logic which is in NP. In our notation, it can easily be seen that $K4B = K(\varphi_{w\geq 0, x\geq 0, y\geq 0}^{1-0}, \varphi_{w\geq 0, x\geq 0, y\geq 0}^{0-2})$. For this concrete example and many generalizations, the following theorem gives this result.

**Theorem 4.15.** Let $k \geq 2 \in \mathbb{N}$, and let $p_1, q_1, r_1, p_2, q_2, r_2 \in \mathbb{N}$. Let $\varphi$ be a universal Horn formula such that $\varphi$ implies $\varphi_{w\geq p_1, x\geq q_1, y\geq r_1}^{1-0} \land \varphi_{w\geq p_2, x\geq q_2, y\geq r_2}^{0-k}$ if $k$ is even. Let $G$ be a graph satisfying $\varphi_{w\geq p_1, x\geq q_1, y\geq r_1}^{1-0}$, and let $w = x_0, x_1, x_2$ if $k$ is even and $w = y_0, \ldots, y_{(k-1)k}$ if $k$ is even be nodes in $G$ such that $(x_i, x_{i+1})$ and $(y_i, y_{i+1})$ are edges for all relevant $i$. First, let $k$ be odd. Since $G \models \varphi_{w\geq p_1, x\geq q_1, y\geq r_1}^{1-0}$, it follows that $G$ is symmetric. By applying the $\varphi_{w\geq p_1, x\geq q_1, y\geq r_1}^{0-k}$-property $k - 1$ times, we get a path of length $k - 1$ from $w$ to $y_{(k-1)k}$. Hence, since $(x_1, w)$ is an edge, it follows that $G \models x_1 \rightarrow y_{(k-1)k}$, and due to the $\varphi_{w\geq p_1, x\geq q_1, y\geq r_1}^{0-k}$-property, this implies that $(x_1, y_{(k-1)k})$ is an edge as required.

Now let $k$ be even. Since $k \equiv 0 \mod 2$, and $k \geq 2$, the symmetry of $G$ ensures that there is a path of length $k$ from $w$ to $x_2$. Hence, there is an edge $(w, x_2)$, and due to the symmetry of $G$, an edge $(x_2, w)$. By $(k-1)$ applications of the $\varphi_{w\geq p_1, x\geq q_1, y\geq r_1}^{0-k}$-property, we know that $G \models w \rightarrow y_{(k-1)k}^{k-1}$. Hence, we conclude that $G \models x_2 \rightarrow y_{(k-1)k}^{k}$, and another application of the $\varphi_{w\geq p_1, x\geq q_1, y\geq r_1}^{0-k}$-property gives the edge $(x_2, y_{(k-1)k})$, as required.

For the NP-result, observe that $\varphi_{w\geq 0, x\geq 0, y\geq 0}^{1-0} \land \varphi_{w\geq 0, x\geq 0, y\geq 0}^{0-k}$ implies $\varphi_{w\geq 0, x\geq 0, y\geq 0}^{0-k}$ and $\varphi_{w\geq 0, x\geq 0, y\geq 0}^{1-0}$ imply $\varphi_{w\geq 0, x\geq 0, y\geq 0}^{0-k}$, where $p = \max(p_1, p_2)$, $q = \max(q_1, q_2)$, and $r = \max(r_1, r_2)$. Due to the above and Lemma 4.11, we know that this formula implies $\varphi_{w\geq 0, x\geq 0, y\geq 0}^{0-k}$ for some $p', q', r'$ if $k$ is odd, and it implies $\varphi_{w\geq 0, x\geq 0, y\geq 0}^{2-(k-1)k}$ for some $p'', q'', r''$ if $k$ is even. Further, if $k$ is odd, then it follows that $k \geq 3$. Hence, $(k-1)k \geq 6$. If $k$ is even, then, since $k \geq 2$, we know that $(k-1)k \geq 4$, and in both cases the NP result follows from Corollary 4.14.

Theorem 4.15 concludes our results about logics defined by specific Horn formulas. We now have collected all tools required to prove the main result of the paper, the complexity classification of satisfiability problems for logics defined by universal Horn formulas.

### 4.5 The Main Result: A Dichotomy for Horn Formulas

In this section, we show a dichotomy theorem, which classifies the complexity of the satisfiability problem for logics of the form $K(\hat{\varphi})$, where $\hat{\varphi}$ is a conjunction of universal Horn
clauses, into solvable in NP and PSPACE-hard. The classification is given in the form of the algorithm HORN-CLASSIFICATION presented in Figure 5. In order to explain the algorithm, we need some more definitions.

For a set \( \text{types-list} \subseteq \{ \text{refl}, \text{symm}, \text{trans}^k \mid k \in \mathbb{N} \} \), we say that a graph \( G \) satisfies the conditions of \( \text{types-list} \) if it has the corresponding properties, i.e., if \( \text{refl} \in \text{types-list} \), then \( G \) is required to be reflexive, if \( \text{symm} \in \text{types-list} \), then \( G \) is required to be symmetric, and if \( \text{trans}^k \in \text{types-list} \), then \( G \) is required to be \( k \)-transitive. A \( \text{types-list} \)-tree is a graph which can be obtained from a strict tree \( T \) by adding exactly those edges required to make it satisfy the conditions of \( \text{types-list} \) (note that this is a natural closure operator). Similarly, a \( \text{types-list} \)-line is the \( \text{types-list} \)-closure of a strict line. Finally, for a universal Horn clause \( \varphi \), let \( \text{types-list} - T^\varphi_{\text{hom}} \) denote the pairs \((\alpha, T)\) such that \( T \) is a \( \text{types-list} \)-tree, and \( \alpha \) \( \text{prereq} (\varphi) \rightarrow T \) is a homomorphism. Intuitively, due to Proposition 4.3, this is the set of \( \text{types-list} \)-trees about which the clause \( \varphi \) makes a statement, along with the corresponding homomorphisms. We first define the cases of universal Horn clauses leading to NP-containment of the satisfiability problem:

**Definition 4.16.** Let \( \varphi \) be a universal Horn clause, and \( \text{types-list} \subseteq \{ \text{refl}, \text{symm}, \text{trans}^k \mid k \in \mathbb{N} \} \). We say that \( (\varphi, \text{types-list}) \) satisfies the NP-case, if one of the following occurs:

1. \( \text{conc} (\varphi) = (x, y) \) for \( x \neq y \in \text{prereq} (\varphi) \), and there is \((\alpha, T) \in \text{types-list} - T^\varphi_{\text{hom}} \) such that there is no directed path connecting \( \alpha(x) \) and \( \alpha(y) \) in \( T \),
2. \( \text{types-list} - T^\varphi_{\text{hom}} \neq \emptyset \), and \( \text{conc} (\varphi) = \emptyset \) or the vertices from \( \text{conc} (\varphi) \) are different and not connected with an undirected path in \( \text{prereq} (\varphi) \) (this also applies if \( x \) or \( y \) do not appear in \( \text{prereq} (\varphi) \))
3. \( \text{conc} (\varphi) = (x, y) \) for \( x, y \in \text{prereq} (\varphi) \), and there is a homomorphism \( \alpha : \text{prereq} (\varphi) \rightarrow L \), where \( L \) is the \( \text{types-list} \)-line \((x_1, \ldots, x_n)\), and there are \( i, j \) such that \( 1 \leq i < j - 2 \) such that \( \alpha(x) = x_j \) and \( \alpha(y) = x_i \),
4. \( \text{conc} (\varphi) = (x, y) \) for \( x, y \in \text{prereq} (\varphi) \), and there exist \( \text{types-list} \)-lines \( L_1 = (x_0, \ldots, x_n) \) and \( L_2 = (y_0, \ldots, y_n) \) and homomorphisms \( \alpha_1 : \text{prereq} (\varphi) \rightarrow L_1 \), \( \alpha_2 : \text{prereq} (\varphi) \rightarrow L_2 \), such that \( \alpha_1(x) = x_{i_1} \), and \( \alpha_1(y) = x_{i_1 - 1} \), \( \alpha_2(x) = y_{i_2} \), and \( \alpha_2(y) = y_{i_2 + k} \), where \( k \geq 2 \).

We now define the properties of Horn clauses which do not lead to NP-containment of the satisfiability problems on their own. Recalling Section 2.2, it is natural that clauses which are satisfied in every reflexive, transitive, or symmetric tree are among these. This is captured by the following definitions: If \( \varphi \) is a universal Horn clause with \( \text{conc} (\varphi) = (x, y) \) for \( x = y \) or \( x, y \in \text{prereq} (\varphi) \), we say that \( (\varphi, \text{types-list}) \) satisfies the reflexive case, if \( x = y \) or for every \((\alpha, T) \in \text{types-list} - T^\varphi_{\text{hom}} \) such that \((\alpha(x), \alpha(y))\) is not an edge in \( T \), it holds that \( \alpha(x) = \alpha(y) \). \((\varphi, \text{types-list}) \) satisfies the transitive case for \( k \in \mathbb{N} \), if it does not satisfy the reflexive case, and for every \((\alpha, T) \in \text{types-list} - T^\varphi_{\text{hom}} \) such that \((\alpha(x), \alpha(y))\) is not an edge in \( T \), there is a path from \( \alpha(x) \) to \( \alpha(y) \) in \( T \), and there is some \((\alpha, T) \in \text{types-list} - T^\varphi_{\text{hom}} \) such that there is no edge \((\alpha(x), \alpha(y))\), and \( \alpha(y) \) is exactly \( k \) levels below \( \alpha(x) \) in \( T \). Finally, \((\varphi, \text{types-list}) \) satisfies the symmetric case if it does not satisfy the reflexive of the transitive case, and for every \((\alpha, T) \in \text{types-list} - T^\varphi_{\text{hom}} \) such that \((\alpha(x), \alpha(y))\) is not an edge in \( T \), there is an edge \((\alpha(y), \alpha(x))\) in \( T \).

We can now state the classification theorem—the proof will follow from the individual results in this section. Note that the algorithm as stated cannot be implemented directly, since it uses tests of the form if a given first-order formula is satisfied in certain infinite classes of graphs, and checks if certain elements are present in the infinite set \( \text{types-list} - T^\varphi_{\text{hom}} \). However, we believe that size-restrictions for the structures actually required to look at can be proven, and hence the algorithm hopefully can be implemented to give a deterministic
decision procedure. However, the main usage of the algorithm is to show more general classification theorems, as we will see in Corollary 4.29.

**Theorem 4.17.** Let $\psi$ be a conjunction of universal Horn clauses. Then the complexity of $K(\psi)$-SAT is correctly determined by HORN-CLASSIFICATION.

A first look at the algorithm in Figure 5 reveals that it is obviously necessary to prove that the choices that the algorithm has to make always can be made: in the relevant situations, at least one of the “reflexive,” “transitive,” or “symmetric” conditions occurs. However, before starting with the proof, we explain the general idea of the algorithm and give an example for a logic which HORN-CLASSIFICATION proves to have a satisfiability problem in NP.

1: $\text{types-list} := \emptyset$
2: while not done do
3: if every clause in $\hat{\psi}$ is satisfied on every $\text{types-list}$-tree then
4: $K(\hat{\psi})$-SAT is PSPACE-hard
5: end if
6: Let $\hat{\varphi}$ be a clause in $\hat{\psi}$ not satisfied on every $\text{types-list}$-tree
7: if $(\hat{\varphi}, \text{types-list})$ satisfies the NP-case then
8: $K(\hat{\psi})$ has the polynomial-size model property, and $K(\hat{\psi})$-SAT $\in$ NP.
9: else
10: conc $(\hat{\varphi}) = (x, y)$ for $x = y$ or $x, y \in \text{prereq} (\hat{\varphi})$
11: if $(\hat{\varphi}, \text{types-list})$ satisfies the reflexive case then
12: $\text{types-list} := \text{types-list} \cup \{\text{refl}\}$
13: else if $(\hat{\varphi}, \text{types-list})$ satisfies the transitive case for $k \geq 2$ then
14: $\text{types-list} := \text{types-list} \cup \{\text{trans}^k\}$
15: else if $(\hat{\varphi}, \text{types-list})$ satisfies the symmetric case then
16: $\text{types-list} := \text{types-list} \cup \{\text{symm}\}$
17: end if
18: end if
19: if for some $k$, $\{\text{symm}, \text{trans}^k\} \subseteq \text{types-list}$ then
20: $K(\hat{\psi})$ has the polynomial-size model property, and $K(\hat{\psi})$-SAT $\in$ NP.
21: end if
22: end while

**Fig. 5.** The algorithm HORN-CLASSIFICATION

In the variable $\text{types-list}$, the algorithm maintains a list of implications of the formula $\psi$. For example, HORN-CLASSIFICATION puts $\text{refl}$ into $\text{types-list}$ if it detects the formula $\psi$ to require a graph satisfying it to be “near-reflexive” (meaning, reflexive in all nodes with sufficient height and depth). Similarly, $\text{symm} \in \text{types-list}$ means that $\psi$ requires a graph to be “near-symmetric,” and $\text{trans}^k \in \text{types-list}$ means that $\psi$ requires a graph to be “near-$k$-transitive” (this will be made precise in Lemma 4.27).

If at one point HORN-CLASSIFICATION detects that a clause $\hat{\varphi}$ satisfies one of the NP-conditions, then this means that the clause $\hat{\varphi}$, in addition with the requirements kept in $\text{types-list}$, implies a graph-property which leads to the polynomial-size model property. It is well-known that the modal logic over the class of frames which are both transitive and symmetric has a satisfiability problem in NP. Generalizing this, when HORN-CLASSIFICATION detects that $\hat{\psi}$ requires “near-symmetry” and “near-$k$-transitivity,” this also leads to the polynomial-size model property and thus to NP-membership of the satisfiability problem, applying Theorem 4.15.
As an example, let \( \bar{\varphi} \) be the universal Horn clause with prerequisite graph as shown in Figure 6, with \( \text{conc}(\bar{\varphi}) = (x, y) \), and let \( \bar{\psi} \) be the Horn formula having \( \bar{\varphi} \) as its only conjunct.

The algorithm \textsc{Horn-Classification} starts with \( \text{types-list} = \emptyset \), and hence in its first iteration, checks if \( \bar{\varphi} \) is satisfied in every strict tree. This is not the case, as Figure 7 shows (here, we simply marked each note with the names of the vertices which are preimages of the homomorphism): This is a homomorphic image of \( \text{prereq}(\bar{\varphi}) \) as a strict line (which in particular is a strict tree), in which the images of \( x \) and \( y \) are not connected with an edge. Therefore, \( \bar{\varphi} \) is not satisfied in this line, and hence not in every strict tree. However, it is clear that if we want to map \( \text{prereq}(\bar{\varphi}) \) into a strict tree, then all the vertices between \( s \) and \( t \) must be “pairwise identified,” just like in Figure 7.

Therefore the transitive case is satisfied, and the figure shows that this is true for \( k = 3 \) (among possibly others). Hence \textsc{Horn-Classification} adds the element \( \text{trans}^3 \) to \( \text{types-list} \). Next it checks if \( \bar{\varphi} \) is satisfied in every \( \{\text{trans}^3\} \)-tree. This again is not the case, and the homomorphic image of \( \text{prereq}(\bar{\varphi}) \) as a \( \{\text{trans}^3\} \)-line in Figure 8 shows that \( \bar{\varphi} \) satisfies NP-condition 3 (we only included those lines added by the \( \text{trans}^3 \)-closure that are required for our function to be a homomorphism). Therefore, the logic \( K(\bar{\varphi}) \) has the polynomial-size model property, and its satisfiability problem is in NP.

This example demonstrates that in the run of \textsc{Horn-Classification}, a clause \( \bar{\varphi} \) can meet different cases depending on the content of the variable \( \text{types-list} \): In the situation that \( \text{types-list} = \emptyset \), the clause \( \bar{\varphi} \) satisfies the transitive case, but when \( \text{types-list} = \{\text{trans}^3\} \), this is no longer the case.

As mentioned above, the first fact that we need to prove about \textsc{Horn-Classification} is that it is well-defined, that is, the case distinction between the “reflexive,” “transitive” and “symmetric” cases is complete.

**Lemma 4.18.** \textsc{Horn-Classification} is well-defined: in all relevant situations, at least one of the “reflexive,” “transitive,” and “symmetric” cases occurs.

**Proof.** Assume that the algorithm is not well-defined, and let \( \bar{\psi} \) be an instance for which the algorithm behavior is unspecified. Let \( \bar{\varphi} \) be a clause for which none of the NP-conditions and none of the reflexive, transitive, and symmetric conditions hold. Let \( \text{types-list} \) be as determined by \textsc{Horn-Classification} when encountering \( \bar{\varphi} \).

Since the algorithm only chooses \( \bar{\varphi} \) if \( \bar{\varphi} \) is not satisfied on every \( \text{types-list}\)-tree, we know that due to Proposition 4.3, there is some \( (\alpha, T) \in \text{types-list} \setminus T^\text{hom}_{\bar{\varphi}} \). Now assume that
conc(\hat{\phi}) = \emptyset or conc(\hat{\phi}) = (x,y) for x \neq y, and x,y do not both appear in \text{prereq}(\hat{\phi}). Then NP-condition 2 is satisfied, and we have a contradiction. Therefore we know that \hat{\phi} is a universal Horn clause such that conc(\hat{\phi}) = (x,y) for some variables x,y such that \(x = y\) or \(x,y \in \text{prereq}(\hat{\phi})\). Since the reflexive case does not apply, we know that \(x \neq y\), and hence \(x,y \in \text{prereq}(\hat{\phi})\).

Since NP-condition 1 does not apply, we know that for every \((\alpha,T) \in \text{types-list} - T^\text{hom}_{\hat{\phi}}\), there is a directed path connecting \(\alpha(x)\) and \(\alpha(y)\).

Since the transitive case does not apply, we know that there is some \((\alpha,T) \in \text{types-list} - T^\text{hom}_{\hat{\phi}}\) in which there is no path from \(\alpha(x)\) to \(\alpha(y)\). We therefore know, by the above, that there is a path from \(\alpha(y)\) to \(\alpha(x)\) in \(T\). Hence, \(T\) is not symmetric. Since \(T\) is a \text{types-list}-tree, we know that \(\text{symm} \notin \text{types-list}\).

Since there is a path from \(\alpha(y)\) to \(\alpha(x)\) in \(T\) but not vice versa, we know that \(\alpha(x) \neq \alpha(y)\), and since there is a path from \(\alpha(y)\) to \(\alpha(x)\) in the (possibly reflexive and/or \(S\)-transitive for some set \(S\)) tree \(T\), this implies that \(\alpha(x)\) is on a lower level in \(T\) than \(\alpha(y)\). Let \(k\) denote the difference in levels between \(\alpha(x)\) and \(\alpha(y)\), then \(k \geq 1\). By mapping \(T\) into a \text{types-list}-line \(L_1 = (x_0, \ldots, x_n)\) with the homomorphism \(\beta\) assigning each vertex \(u\) the element \(x_i\), where \(i\) is the level of \(u\) in \(T\), we can, using the homomorphism \(\alpha_1 := \beta \circ \alpha\), map \(\text{prereq}(\hat{\phi})\) homomorphically into \(L_1\) in such a way that \(\alpha_1(x) = x_i\), and \(\alpha_1(y) = x_{i-k}\), where \(i\) denotes the level of \(\alpha(x)\) in \(T\).

If \(k \geq 2\), then the line \(L_1\) satisfies NP-condition 3, a contradiction. Therefore, we know that \(k = 1\), and thus \(\alpha_1\) and \(L_1\) satisfy NP-condition 4.

Since \(\hat{\phi}\) also does not satisfy the symmetric case, we know that there is some \((\alpha,T) \in \text{types-list} - T^\text{hom}_{\hat{\phi}}\) such that there is no edge \((\alpha(x),\alpha(y))\) and no edge \((\alpha(y),\alpha(x))\) in \(T\). By NP-condition 1, we know that there is a directed path connecting \(\alpha(x)\) and \(\alpha(y)\) in \(T\).

Since \(\text{symm} \notin \text{types-list}\), and \(T\) is a \text{types-list}-tree, we know that \(T\) is not symmetric. If \(\text{refl} \in \text{types-list}\), and \(T\) is a \text{types-list}-tree, this implies that \(T\) is reflexive, and hence \(\alpha(x) \neq \alpha(y)\). If \(\text{refl} \notin \text{types-list}\), then we know that, since \(\text{symm}\) also is not an element of \text{types-list}, that no node in \(T\) is connected to itself with a directed path, and therefore we also know that \(\alpha(x) \neq \alpha(y)\). Now assume that there is a path from \(\alpha(y)\) to \(\alpha(x)\) in \(T\). Then the shortest of these paths must have length \(k \geq 2\) (since there is no edge \((\alpha(y),\alpha(x))\) and therefore we can map \(T\) (and therefore \(\text{prereq}(\hat{\phi})\)) homomorphically into a line \(L_1\) as in the case above, where the distance of the image of \(\alpha(x)\) and \(\alpha(y)\) is \(k\), and since \(\alpha(y)\) is mapped to a predecessor of \(\alpha(x)\), this line satisfies NP-condition 3, a contradiction. Therefore we know that in \(T\), there is a directed path from \(\alpha(x)\) to \(\alpha(y)\). Since \((\alpha(x),\alpha(y))\) is no edge in \(T\), we know that the shortest of these paths must have length \(\geq 2\). Since \(T\) is not symmetric, this implies that \(\alpha(y)\) must be on a lower level in \(T\) than \(\alpha(x)\), and the difference in levels is \(\geq 2\). Similarly, to the above, we can construct a \text{types-list}-line \(L_2 = (x_0, \ldots, x_m)\) and a homomorphism \(\beta: T \to L_2\), such that for \(v \in T\), if \(i\) is the level of \(v\) in \(T\), then \(\beta(v) = x_i\). Let \(\alpha_2\) denote the homomorphism \(\beta \circ \alpha\). Due to the choice of \(k\) and the definition of \(\beta\), we know that \(\alpha_2(x) = x_i\), and \(\alpha_2(y) = x_{i+k}\) for some \(i\), and hence \(\alpha_2\) and \(L_2\) satisfy NP-condition 4.

Since above, we already constructed \(\alpha_1\) and \(L_1\) satisfying NP-condition 4, this implies that \(\hat{\phi}\) satisfies NP-condition 4, a contradiction.

Also note that in the transitive case, a \(k \geq 2\) can always be chosen: Since by construction of the algorithm, \(\hat{\phi}\) is not satisfied on every \text{types-list}-tree, there is some \((\alpha,T)\) such that \((\alpha(x),\alpha(y))\) is not an edge in \(T\). Since the reflexive case does not apply, we can choose \((\alpha,T)\) in such a way that \(\alpha(x) \neq \alpha(y)\). Due to the transitive case, there is a path from \(\alpha(x)\) to \(\alpha(y)\) in \(T\), and hence the shortest of these paths must have length of at least 2, hence the difference of levels is at least 2 (we know that even if \(\text{symm} \in \text{types-list}\), \(\alpha(y)\) must be below \(\alpha(x)\) in \(T\), otherwise NP-condition 3 would apply). \(\square\)
Now that we know that HORN-CLASSIFICATION is well-defined, we show that it always comes to a halt and hence, generates an answer—this is not immediate: Note that in the example formula from Figure 6 discussed above, we already saw that it is possible for the algorithm to revisit a clause. Therefore one might consider it possible for the algorithm to repeatedly add the same element to the variables types-list without coming to a halt, or the list types-list to grow infinitely. In order to show that this does not happen, we prove a bound on the number of the elements that can be added into this list during the run of the algorithm.

A first helpful tool to show this is the following Proposition, which says that there are no “redundant” transitivity conditions which get added to types-list.

**Proposition 4.19.** Let $\text{trans}^k$ be added to the variable types-list by HORN-CLASSIFICATION. Then types-list did not imply $k$-transitivity before adding $\text{trans}^k$.

**Proof.** Assume that types-list already implied $\text{trans}^k$. Since $\text{trans}^k$ is added by HORN-CLASSIFICATION, there is a clause $\phi$ with conclusion edge $(x, y)$ for $x, y \in \text{prereq}(\phi)$ which satisfies the transitive condition for $k$ at this point of the algorithm’s run. Therefore, there is some $(\alpha, T) \in \text{types-list}^{\text{hom}}$ such that $(\alpha(x), \alpha(y))$ is not an edge in $T$, and $\alpha(y)$ is exactly $k$ levels below $\alpha(x)$. Since by the assumption the conditions of types-list already imply $k$-transitivity, the $k$-step path from $\alpha(x)$ to $\alpha(y)$ implies that $(\alpha(x), \alpha(y))$ is an edge in $T$, a contradiction. □

Now that we know that transitivity conditions which were already implied by the conditions present in types-list do not get added to the list, there is a clear strategy how we can prove that types-list does not grow infinitely: We first prove just how many transitivity conditions are implied by the conditions in types-list, and then show that from a certain point on, everything that will be added to types-list by HORN-CLASSIFICATION already is implied. In order to prove this, we need some technical results about implications of various forms of $S$-transitivity.

**Lemma 4.20.** Let $k_1, k_2 \in \mathbb{N}$. Then $\{k_1, k_2\}$-transitivity implies $(k_2 + l \cdot (k_1 - 1))$-transitivity for all $l \geq 0$.

**Proof.** We show the claim by induction on $l$. For $l = 0$, this holds trivially, since $\{k_1, k_2\}$-transitivity by definition implies $k_2$-transitivity. Now assume that it holds for $l$, let $G$ be some graph which is $\{k_1, k_2\}$-transitive, and let $u_0, \ldots, u_{k_2 + (l+1) \cdot (k_1 - 1)}$ be nodes in $G$ such that $(u_i, u_{i+1})$ is an edge for all relevant $i$. We need to show that $(u_0, u_{k_2 + (l+1) \cdot (k_1 - 1)})$ is an edge in $G$. Since by the induction hypothesis, $\{k_1, k_2\}$-transitivity implies $(k_2 + l \cdot (k_1 - 1))$-transitivity, we know that $G$ is $(k_2 + l \cdot (k_1 - 1))$-transitive. Therefore there is an edge $(u_0, u_{k_2 + (l \cdot (k_1 - 1))})$ in $G$. By choice of $u_i$, there is a $(k_1 - 1)$-step path from $u_{k_2 + (l \cdot (k_1 - 1))}$ to $u_{k_2 + (l+1) \cdot (k_1 - 1)}$. Therefore there is a $k_1$-transitive, this implies that $(u_0, u_{k_2 + (l+1) \cdot (k_1 - 1)})$ is an edge in $G$, as required. □

**Lemma 4.21.** Let $m, k_1, e_1, d \in \mathbb{N}$ such that $(k_1 - 1) = e_1 \cdot d$. Then $\{k_1, m, m + d, \ldots, m + (e_1 - 1) \cdot d\}$-transitivity implies $(m + l \cdot d)$-transitivity for all $l \geq 0$.

**Proof.** Let $l = p \cdot e_1 + r$ for some $p, r \in \mathbb{N}$, $r < e_1$. Then $m + l \cdot d = m + (p \cdot e_1 + r) \cdot d = m + r \cdot d + p \cdot e_1 \cdot d = m + r \cdot d + p(k_1 - 1)$. Since $r < e_1$, we know that $\{k_1, m, m + d, \ldots, m + (e_1 - 1) \cdot d\}$-transitivity implies $(m + r \cdot d)$-transitivity. By Lemma 4.20, we know that $\{k_1, m + r \cdot d\}$-transitivity implies $(m + r \cdot d + p(k_1 - 1))$-transitivity, and since $k_1$-transitivity is obviously implied by the prerequisites, this proves the Lemma. □
Lemma 4.22. Let \((k_1 - 1) = e_1 \cdot d, \) and let \(n_0, \ldots, n_{e_1-1} \in \mathbb{N}\) such that \(n_i \equiv n_j \mod d, \) and for \(i \neq j, n_i \neq n_j \mod (k_1 - 1). \) Then there exists some \(m \in \mathbb{N}\) such that \(m \equiv n_i \mod d\) such that \(\{k_1, n_0, \ldots, n_{e_1-1}\}\)-transitivity implies \((m + l \cdot d)\)-transitivity for all \(l \geq 0.\)

Proof. Let \(S := \{k_1, n_0, \ldots, n_{e_1-1}\}, \) and let \(n_i = a_i \cdot (k_1 - 1) + n_i', \) where \(n_i' < k_1 - 1.\) Due to Lemma 4.20, \(\{k_1, n_i\}\)-transitivity implies \((n_i + l \cdot (k_1 - 1))\)-transitivity for all \(l \geq 0.\)

Therefore we can choose \(a := \max \{a_0, \ldots, a_{e_1-1}\}, \) and we know that \(S\)-transitivity implies \((a \cdot (k_1 - 1) + n_i')\)-transitivity. Since for all \(a \), \(n_i + l \cdot (k_1 - 1)\) is equivalent to \(n_i \) modulo \(k_1 - 1\) and modulo \(d\) (since \(d\) divides \(k_1 - 1\)), we can assume without loss of generality that \(n_i = a \cdot (k_1 - 1) + n_i'\) for all \(0 \leq i \leq e_1 - 1.\)

Let \(n_i' = t_i \cdot d + r, \) where \(r < d.\) Such numbers \(t_i\) exist, since \(n_i \equiv n_j \mod d\) for all \(i, j.\)

Since \(n_i' < k_1 - 1,\) we know that \(t_i \cdot d + r < k_1 - 1 = e_1 \cdot d,\) and therefore \(t_i \leq e_1 - 1\) for all \(0 \leq i \leq e_1 - 1.\)

Define \(m := a \cdot (k_1 - 1) + r.\) Note that since \(d\) divides \(k_1 - 1,\) this implies that \(m \equiv r \mod d.\)

For the \(n_i\) it holds that \(n_i = a \cdot (k_1 - 1) + n_i' \) and again, since \(d\) divides \(k_1 - 1,\) we have that \(n_i \equiv n_i' \mod d.\) Now since \(n_i' = t_i \cdot d + r,\) it follows that \(n_i'\) is equivalent to \(r\) modulo \(d,\) and hence \(m\) and \(n_i\) are equivalent modulo \(d\) for all \(i.\) Now note that

\[n_i - m = a \cdot (k_1 - 1) + n_i' - a \cdot (k_1 - 1) - r = n_i' - r = t_i \cdot d + r - r = t_i \cdot d.\]

Therefore, since \(S\)-transitivity implies \(n_i\text{-transitivity for all }i,\) we know that \(S\)-transitivity implies \((m + t_i \cdot d)\)-transitivity for all \(i.\) Since \(t_i \leq e_1 - 1,\) and for \(i \neq j,\) it also holds that \(t_i \neq t_j\) (otherwise it would follow that \(n_i' = n_j'\) and hence \(n_i = n_j,\) a contradiction), it follows that \(\{t_0, \ldots, t_{e_1-1}\} = \{0, \ldots, e_1 - 1\}.\) We therefore know that for each \(i \in \{0, \ldots, e_1 - 1\}, S\)-transitivity implies \((m + i \cdot d)\)-transitivity, and since \(k_1 \in S,\) the claim follows from Lemma 4.21.

The following lemma is the final of our results about implied transitivity conditions. Its technical formulation hides the fact that the statement of the lemma is actually rather natural. As an example, consider the case where \(k_1 = 5,\) and \(k_2 = 7.\) Then \(\gcd(k_1 - 1, k_2 - 1) = 2,\) and then the lemma says that there is some odd number \(m\) such that for all odd numbers \(m + 2 \cdot l, \) \(\{5, 7\}\)-transitivity implies \((m + 2 \cdot l)\)-transitivity. The key idea is that then, once \(\text{trans}^5\) and \(\text{trans}^7\) are elements of \(\text{types-list,}\) we know that due to Proposition 4.19, \(\text{HORN-CLASSIFICATION}\) does not add any more \(\text{trans}^k\)-conditions to \(\text{types-list}\) anymore for odd \(k\)s which are greater than or equal to \(m.\) Therefore there are only finitely many odd \(k\)s such that \(\text{HORN-CLASSIFICATION}\) can add \(\text{trans}^k\) from this point on.

If \(\text{HORN-CLASSIFICATION}\) would add an infinite number of \(\text{trans}^k\)-conditions, then one of them must therefore be one where \(k\) is even. Assume that this already happens with \(k_3,\) i.e., that \(k_3\) is an even number. Then \(k_3 - 1\) is odd, and therefore the greatest common divisor of \(k_1 - 1, k_2 - 1,\) and \(k_3 - 1\) is 1. By the lemma, we therefore know that there is some \(m\) such that \(\{5, 7, k_3\}\)-transitivity implies \(m'\text{-transitivity for all }m' \geq m,\) and then \(\text{HORN-CLASSIFICATION}\) can only add \(\text{trans}^k\)-conditions for \(k \leq m',\) and this is only a finite number of possibilities. Thus we know that \(\text{HORN-CLASSIFICATION}\) must stop adding transitivity conditions at some point.

With the above proof-strategy in mind, it is clear that the statement of the following lemma is crucial in restricting the number of \(\text{trans}^k\)-conditions added by \(\text{HORN-CLASSIFICATION.}\)

Lemma 4.23. Let \(k_1, \ldots, k_n \in \mathbb{N},\) and let \(d := \gcd((k_1 - 1), \ldots, (k_n - 1)).\) Then there exists some \(m \in \mathbb{N}\) such that \(m \equiv 1 \mod d\) and \(\{k_1, \ldots, k_n\}\)-transitivity implies \((m + l \cdot d)\)-transitivity for all \(l \geq 0.\)
Proof. Let \( S = \{k_1, \ldots, k_n\} \). We show the claim inductively on \( n \). If \( n = 1 \), then \( d = k_1 - 1 \), and by Lemma 4.20, \( \{k_1, k_2\} \)-transitivity implies \( (k_1 + l \cdot (k_1 - 1)) \)-transitivity for all \( l \geq 0 \), hence the claim follows with \( m = k_1 \).

Since we need the case \( n = 2 \) explicitly in the induction, we prove it individually. Hence let \( n = 2 \), and let \( (k_1 - 1) = e_1 \cdot d \), where \( \gcd(e_1, e_2) = 1 \).

From Lemma 4.20, we know that \( \{k_1, k_2\} \)-transitivity implies \( k_1 + i \cdot (k_2 - 1) = i \cdot k_2 + k_1 - i \)-transitivity for all \( i \geq 0 \). Now note that for all \( 0 \leq i \leq e_1 - 1 \):

\[
i \cdot k_2 + (k_1 - i) = i \cdot (k_2 - 1) + (k_1 - 1) + 1 = i \cdot e_2 \cdot d + e_1 \cdot d + 1 = d \cdot (i \cdot e_2 + e_1) + 1 =; n_i.
\]

Due to Lemma 4.22, it suffices to show that all of these \( n_i \) are equivalent to 1 modulo \( d \), and for \( 0 \leq i < j \leq e_1 - 1 \), \( n_i \not\equiv n_j \mod (k_1 - 1) \). It is obvious that all \( n_i \) are equivalent to 1 modulo \( d \). Now assume that there are \( 0 \leq i < j \leq e_1 - 1 \) such that \( n_i \equiv n_j \mod (k_1 - 1) \). Then the following holds:

\[
\begin{align*}
d \cdot (j \cdot e_2 + e_1) + 1 & \equiv d \cdot (i \cdot e_2 + e_1) + 1 \mod (k_1 - 1) \\
d \cdot (j \cdot e_2 + e_1) & \equiv d \cdot (i \cdot e_2 + e_1) \mod (k_1 - 1) \\
d \cdot (j \cdot e_2 - i \cdot e_2) & \equiv 0 \mod (k_1 - 1) \\
d \cdot e_2 \cdot (j - i) & \equiv 0 \mod (k_1 - 1) \\
d \cdot e_2 \cdot (j - i) & = t \cdot d \cdot e_1 \quad \text{for some } t \\
e_2 \cdot (j - i) & \equiv 0 \mod e_1 \\
j - i & \equiv 0 \mod e_1, \text{ since } \gcd(e_1, e_2) = 1
\end{align*}
\]

This is a contradiction, since due to the above, we know that \( 0 < j - i \leq e_1 - 1 \). This completes the proof for the case \( n = 2 \).

Now inductively assume that the lemma holds for \( n \geq 2 \). Let \( d' := \gcd(k_1 - 1, \ldots, k_n - 1) \). Recall that \( d = \gcd(k_1 - 1, \ldots, k_n - 1) \). We first show that \( d = \gcd(k_{n+1} - 1, d') \), which is a standard fact from number theory. Obviously, \( d \) divides both \( k_{n+1} \) and \( d' \), and hence \( d \mid \gcd(k_{n+1} - 1, d') \). On the other hand, let \( g \) be a common divisor of \( k_{n+1} - 1 \) and \( d' \). Since \( g \) divides \( d' \), \( g \) is also a divisor of \( k_{n+1} - 1 \). Since \( g \) divides \( k_{n+1} - 1 \), it is a common divisor of all \( k_1 - 1 \). Therefore, \( g \mid d \). In particular, this holds for \( g = \gcd(k_{n+1} - 1, d') \), and therefore \( \gcd(k_{n+1} - 1, d') \mid d \). Since the other direction holds due to the above, we have shown that \( d = \gcd(k_{n+1} - 1, d') \).

Due to the induction hypothesis, we know that there is a natural number \( m' \) such that \( m' \equiv 1 \mod d' \), and for all \( l' \geq 0 \), \( S \)-transitivity implies \( (m' + l' \cdot d') \)-transitivity. Let \( m' = q \cdot d' + 1 \), and let \( l' \) be a natural number such that \( \gcd(q + l', k_{n+1} - 1) = 1 \). Then, by choice of \( m' \), we know that \( S \)-transitivity implies \( (m' + l' \cdot d') \)-transitivity. Let \( e := \gcd(m' + l' \cdot d' - 1, k_{n+1} - 1) \). Since \( S \)-transitivity also implies \( k_{n+1} \)-transitivity, we know from the case \( n = 2 \) that there exists a natural number \( m \) such that \( m \equiv 1 \mod e \), and for all \( l \geq 0 \), \( S \)-transitivity implies \( (m + l \cdot e) \)-transitivity. In order to prove the lemma, it therefore suffices to show that \( e = d \). It holds that

\[
e = \gcd(m' + l' \cdot d' - 1, k_{n+1} - 1) = \gcd(q \cdot d' + 1 + l' \cdot d' - 1, k_{n+1} - 1) = \gcd((q + l') \cdot d', k_{n+1} - 1).
\]

Since \( d \) is a divisor of \( d' \) and of \( k_{n+1} - 1 \), it follows that \( d \mid e \). On the other hand, \( e \) is a common divisor of \( (q + l') \cdot d' \) and \( k_{n+1} - 1 \). Since \( \gcd(q + l', k_{n+1} - 1) = 1 \), we know that \( e \) divides \( d' \). Since \( e \) also divides \( k_{n+1} - 1 \), this implies that \( e \) is a divisor of \( \gcd(d', k_{n+1} - 1) = d \). Therefore \( e \mid d \) and \( d \mid e \), and hence we have proven that \( e = d \), as claimed. \( \square \)
We can now prove that there are only finitely many additions of the form \( \text{trans}^k \) during the algorithm’s run, using the arguments from the discussion before Lemma 4.23:

**Lemma 4.24.** Let \( \hat{\psi} \) be a universal Horn formula. Then HORN-CLASSIFICATION, on input \( \hat{\psi} \), only adds finitely many conditions of the form \( \text{trans}^k \) to types-list.

**Proof.** Assume that this is not the case. Due to Proposition 4.19, we know that each added \( \text{trans}^k \)-element is not implied by the previous elements, in particular, no element is added twice. Hence there is an infinite sequence \( (k_n)_{n \in \mathbb{N}} \) such that all \( \text{trans}^k \) are added to types-list, they are added in this order, and for \( n \neq m \), we have \( k_n \neq k_m \).

For \( n \in \mathbb{N} \), let \( d_n := \gcd(k_1 - 1, \ldots, k_n - 1) \). Then \( d_n \) is obviously decreasing, and bounded by 1. Therefore, the sequence converges, i.e., there is some \( d, n_0 \in \mathbb{N} \) such that \( d_n = d \) for all \( n \geq n_0 \). Due to Lemma 4.23, we know that there is some \( m \) such that \( \{k_1, \ldots, k_{n_0}\}\)-transitivity implies \( (m + l \cdot d)\)-transitivity for all \( l \geq 0 \), and \( m \equiv 1 \mod d \). Since the sequence \( (k_n)_{n \in \mathbb{N}} \) is infinite and no element is repeated, there is some \( n \geq n_0 \) such that \( k_n \geq m \). Since \( \gcd(k_1 - 1, \ldots, k_{n_0} - 1, \ldots, k_n - 1) = d_n = d \), we know that \( d \) divides \( k_n - 1 \). Therefore, let \( k_n = d \cdot p + 1 \). Since \( m \equiv 1 \mod d \), let \( m = d \cdot q + 1 \). Since \( k_n \geq m \), we know that \( p \geq q \). Hence it follows that

\[
 k_n = d \cdot p + 1 = d(p - q + q) + 1 = d \cdot (p - q) + d \cdot q + 1 = d \cdot (p - q) + m.
\]

Due to the choice of \( m \) and since \( p - q \geq 0 \), we therefore know that \( \{k_0, \ldots, k_{n_0}\}\)-transitivity implies \( k_n\)-transitivity. This is a contradiction to Proposition 4.19, since \( k_n \) is added after \( k_0, \ldots, k_{n_0} \). Therefore, there are only finitely many elements of the form \( \text{trans}^k \) added by the algorithm. \( \square \)

Since we have now restricted the number of elements \( \text{trans}^k \) added to types-list, we can prove that the algorithm halts on any input.

**Lemma 4.25.** HORN-CLASSIFICATION always halts.

**Proof.** Assume that it does not halt for an instance \( \hat{\psi} \). Due to Lemma 4.24, we know that there are only finitely many elements of the form \( \text{trans}^k \) which are added to types-list by HORN-CLASSIFICATION. Since the only other elements which can be added to types-list are \( \text{refl} \) and \( \text{symm} \), this implies that there are only finitely many elements of any type which are added to types-list. Now let types-list be as determined by the algorithm. Since the size of types-list is bounded and the variable never shrinks, this is well-defined.

Since the algorithm does not halt, we know that there is a clause \( \hat{\varphi} \) in \( \hat{\psi} \) which is not satisfied in every types-list-tree, and \( \hat{\varphi} \) does not satisfy any of the 4 NP-conditions.

Thus there is some \( (\alpha, T) \in \text{types-list} - T^\text{hom}_{\hat{\varphi}} \) such that \( \hat{\varphi} \) is not satisfied in \( T \). Since we already showed that HORN-CLASSIFICATION is well-defined, and none of the NP-cases applies (otherwise the algorithm would come to a halt), we know that one of the reflexive, transitive, or symmetric cases applies. In particular, \( \text{conc} (\hat{\varphi}) = (x, y) \), and \( x = y \) or \( x, y \in \text{prereq} (\hat{\varphi}) \) and \( (\alpha(x), \alpha(y)) \) is not an edge in \( T \). If one of the cases corresponding to an element not in types-list applies, then this leads to an enlargement of types-list, a contradiction to the choice of types-list. Hence one of the cases corresponding to one of the elements already in types-list applies. We make a case distinction:

**Case 1: the reflexive case applies.** In the case that \( x = y \), the clause is trivially satisfied in the reflexive graph \( T \). Assume that \( x \neq y \). In this case, \( \alpha(x) = \alpha(y) \). But since \( \text{refl} \in \text{types-list} \), and \( (\alpha, T) \in \text{types-list} - T^\text{hom}_{\hat{\varphi}} \), we know that \( T \) is a reflexive tree. Hence, the edge \( (\alpha(x), \alpha(y)) \) exists in \( T \), a contradiction.
Case 2: the transitive case applies for some $k$. In this case, there is a path of length $k$ from $\alpha(x)$ to $\alpha(y)$ in $T$. But since $\text{trans}^k \in \text{types-list}$, and $(\alpha, T) \in \text{types-list} - T^\text{hom}$, we know that $T$ is a $k$-transitive tree. Hence, the edge $(\alpha(x), \alpha(y))$ exists in $T$, a contradiction.

Case 3: the symmetric case applies. In this case, there is an edge from $\alpha(y)$ to $\alpha(x)$ in $T$. But since $\text{symm} \in \text{types-list}$, and $(\alpha, T) \in \text{types-list} - T^\text{hom}$, we know that $T$ is a symmetric tree. Hence, the edge $(\alpha(x), \alpha(y))$ exists in $T$, a contradiction.

Since we have a contradiction in each case, this completes the proof. \qed

Now that we know that the algorithm is both well-defined and comes to a halt, it remains to prove its correctness. The case in which the algorithm states PSPACE-hardness is easily seen to be correct:

**Lemma 4.26.** If for a universal Horn formula $\hat{\psi}$, HORN-CLASSIFICATION states that $\mathcal{K}(\hat{\psi})$-SAT is PSPACE-hard, then this is true.

**Proof.** The only possibility for the algorithm to state that the problem is PSPACE-hard is when the WHILE-loop does not discover any clauses $\hat{\varphi}$ anymore which are not satisfied in every types-list-tree, and types-list does not contain both symm and trans$^k$ for any $k \in \mathbb{N}$. Since each clause $\hat{\varphi}$ in $\hat{\psi}$ is satisfied in every types-list-tree, this implies that the conjunction $\hat{\psi}$ is also satisfied in each types-list-tree. Hence one of the following cases occurs:

- $\hat{\psi}$ is satisfied in every strict tree (if types-list $= \emptyset$),
- $\hat{\psi}$ is satisfied in every reflexive tree (if types-list $= \{\text{refl}\}$),
- there is a set $S \subseteq \mathbb{N}$ such that $\hat{\psi}$ is satisfied in every $S$-transitive tree (if types-list $= \{\text{trans}^k \mid k \in S\}$),
- $\hat{\psi}$ is satisfied in every symmetric tree (if types-list $= \{\text{symm}\}$),
- $\hat{\psi}$ is satisfied in every tree which is both reflexive and symmetric (if types-list $= \{\text{refl, symm}\}$),
- there is a set $S \subseteq \mathbb{N}$ such that $\hat{\psi}$ is satisfied in every tree which is both reflexive and $S$-transitive (if types-list $= \{\text{refl, trans}^k \mid k \in S\}$).

In each of these cases, the hardness result follows directly from Theorem 2.8. \qed

We are now interested in the NP-cases. We first show that if HORN-CLASSIFICATION adds one of the requirements symm, refl, or trans$^k$ for some $k$ to the list types-list, then the formula $\hat{\psi}$ requires any graph $G$ satisfying $\hat{\psi}$ to have the corresponding property (except for vertices with insufficient depth or height in the graph). More precisely, we show the following Lemma:

**Lemma 4.27.** Let $\hat{\psi}$ be a universal Horn formula, and let type $\in \{\text{refl, symm, trans}^k \mid k \in \mathbb{N}\}$ be added to types-list by HORN-CLASSIFICATION on input $\hat{\psi}$. Then the following holds:

- If type $= \text{symm}$, then $\hat{\psi}$ implies $\hat{\varphi}^{1-0}_{w \geq p, x \geq p, y \geq p}$ for some $p \in \mathbb{N}$,
- If type $= \text{refl}$, then $\hat{\psi}$ implies $\hat{\varphi}^{0-0}_{w \geq p, x \geq p, y \geq p}$ for some $p \in \mathbb{N}$,
- If type $= \text{trans}^k$, then $\hat{\psi}$ implies $\hat{\varphi}^{0-k}_{w \geq p, x \geq p, y \geq p}$ for some $p \in \mathbb{N}$.

**Proof.** Inductively assume that all values added before type (if any) were “correct” in the sense that they satisfy the conditions of the lemma. Let the content of the variable types-list directly before adding type be denoted with prev-types-list. Let $\hat{\varphi}$ be the clause in $\psi$ for
which type was added. Then, by the construction of the algorithm, we know that \( \dot{\varphi} \) satisfies the case in the algorithm corresponding to type, it particular, \( \text{conc}(\dot{\varphi}) = (x, y) \) for some variables \( x = y \) or \( x, y \in \text{prereq}(\dot{\varphi}) \). First assume that \( x, y \in \text{prereq}(\dot{\varphi}) \).

We also know that \( \dot{\varphi} \) is not satisfied in every prev-types-list-tree. Hence, let \( (\alpha, T) \in \text{prev-types-list} - T^\text{hom} \) such that \( (\alpha(x), \alpha(y)) \) is not an edge in \( T \). This must exist due to Proposition 4.3. By construction of Horn-Classification, in the case that \( \text{type} = \text{refl} \), we have that \( \alpha(x) = \alpha(y) \), and in the case that \( \text{type} = \text{trans}^k \), we can choose \( (\alpha, T) \) in such a way that \( \alpha(y) \) is is \( k \) levels below \( \alpha(x) \) in \( T \). Finally, in the case that \( \text{type} = \text{symm} \), we know that there is an edge \( (\alpha(y), \alpha(x)) \) in \( T \). Since the reflexive condition does not apply, we can choose \( (\alpha, T) \) in such a way that \( \alpha(x) \neq \alpha(y) \), and since \( \text{symm} \notin \text{prev-types-list} \), this implies that \( \alpha(x) \) is in a lower level of \( T \) than \( \alpha(y) \).

Since all additions before type were correct, we know that all the types in prev-types-list have been added correctly, i.e., we can assume that there is some \( p \) such that

- If \( \text{symm} \in \text{prev-types-list} \), then \( \hat{\psi} \) implies \( \varphi^{1-0}_{w \geq p, x \geq p, y \geq p} \) for some \( p \in \mathbb{N} \),
- If \( \text{refl} \in \text{prev-types-list} \), then \( \hat{\psi} \) implies \( \varphi^{0-0}_{w \geq p, x \geq p, y \geq p} \) for some \( p \in \mathbb{N} \),
- If \( \text{trans}^k \in \text{prev-types-list} \), then \( \hat{\psi} \) implies \( \varphi^{0-k}_{w \geq p, x \geq p, y \geq p} \) for some \( p \in \mathbb{N} \).

We can use the same value \( p \) for all formulas here by taking the maximum, since clearly, \( \varphi^{k-1}_{w \geq p, x \geq q, y \geq r} \) implies \( \varphi^{k-1}_{w \geq p', x \geq q', y \geq r} \) if \( p' \geq p, q' \geq q, \) and \( r' \geq r \). Now let \( L = (x_0, \ldots, x_n) \) be the homomorphic image of \( T \) under the homomorphism \( \delta \) as a prev-types-list-line, where \( \delta \) assigns each \( v \in T \) the node \( x_i \), with \( i \) being the level of \( v \) in \( T \). Let \( \alpha(x) \) be in the another level of \( T \), and let \( \alpha(y) \) be in the \( n - b \)th level of \( T \) (where \( n \) is the height of \( T \)). Due to the construction of \( \delta \), we know that \( \delta(\alpha(x)) = x_a \), and \( \delta(\alpha(y)) = x_{n-b} \). First note that:

- If \( \text{type} = \text{refl} \), then \( \alpha(x) = \alpha(y) \), and hence \( \delta(\alpha(x)) = \delta(\alpha(y)) \). It follows that \( n = a + b \).
- If \( \text{type} = \text{trans}^k \), then by choice of \( T \), \( \alpha(y) \) is exactly \( k \) levels below \( x \) in \( T \), and therefore we know that \( a + b + k = n \).
- If \( \text{type} = \text{symm} \), then, since \( \text{symm} \notin \text{prev-types-list} \), we know that in \( L \), we only have edges between non-decreasing nodes. Since the symmetric case applies, we know that \( (\alpha(y), \alpha(x)) \) is an edge in \( T \). Since \( \alpha(x) \neq \alpha(y) \), \( \alpha(x) \) is at a lower level than \( \alpha(y) \) in \( T \), and it follows that \( a > n - b \). If \( a > n - b + 2 \), then it follows that the line \( L \) satisfies NP-condition 3, a contradiction. Therefore we know that \( a = n - b + 1 \).

We now show that

\[
\begin{align*}
\hat{\psi} & \text{ implies } \varphi^{0-0}_{w \geq a+b+p, x \geq a+b+p, y \geq a+b+p} \quad \text{if } \text{type} = \text{refl}, \\
\hat{\psi} & \text{ implies } \varphi^{0-k}_{w \geq a+b+p, x \geq a+b+p, y \geq a+b+p+k} \quad \text{if } \text{type} = \text{trans}^k, \\
\hat{\psi} & \text{ implies } \varphi^{1-0}_{w \geq a+b+p, x \geq a+b+p+1, y \geq a+b+p} \quad \text{if } \text{type} = \text{symm}.
\end{align*}
\]

For this, let \( G \) be a graph such that \( G \models \hat{\psi} \), by induction we know that the nodes of \( G \) which have both a \( p \)-step predecessor and a \( p \)-step successor satisfy the conditions of prev-types-list. To prove the claim, let \( u, v \) be elements of \( G \) which have both an \( a+b+p \)-step predecessor and an \( a+b+p \)-step successor, and

- \( u = v \) if \( \text{type} = \text{refl} \),
- There is a \( k \)-step path in \( G \) from \( u \) to \( v \) if \( \text{type} = \text{trans}^k \),
- There is an edge \((u, v)\) in \( G \) if \( \text{type} = \text{symm} \).

In order to prove the lemma, we need to prove that \((u, v)\) is an edge in \( G \). Since \( G \models \hat{\psi} \), and \( \hat{\varphi} \) is a clause in \( \hat{\psi} \), it suffices to prove that there is a homomorphism \( \gamma : \text{prereq}(\hat{\varphi}) \to G \)}
such that $\gamma(x) = u$ and $\gamma(y) = v$. Then there is an edge $(u, v)$ in $G$ due to Proposition 4.3. Since $\delta \circ \alpha$: prereq $(\hat{\phi}) \rightarrow L$ is a homomorphism and $\delta \circ \alpha(x) = x_a$ and $\delta \circ \alpha(y) = x_{n-b}$, it suffices to show that there is a homomorphism $\beta: L \rightarrow G$ such that $\beta(x_a) = u$ and $\beta(x_{n-b}) = v$. Then the homomorphism $\gamma := \beta \circ \delta \circ \alpha$ satisfies the necessary conditions. Let $w$ be an $a$-step predecessor of $u$ in $G$ such that $w$ has a $p$-step predecessor in $G$, and let $t$ be a $b$-step successor of $v$ in $G$ which has a $p$-step successor in $G$. Both must exist due to the choice of $u, v$. Additionally, if $\text{types} = \text{symm}$ and therefore $(v, u)$ is an edge in $G$, choose $w$ to be an $a - 1$-step predecessor of $v$ (in which case it is also a $a$-step predecessor of $u$), and in this case also let $t$ to be a $b - 1$-step successor of $u$ (in which case it is also a $b$-step successor of $v$).

Now, define $y_0 := w$, and let $y_1, \ldots, y_n$ from $G$ be chosen in such a way that $y_n = u$, $y_{n-b} = v$, and $y_n = t$, and for all relevant $i$, $(y_i, y_{i+1})$ is an edge in $G$. This is possible since $u$ and $v$ satisfy the conditions corresponding to $\text{type}$:

- If $\text{type} = \text{refl}$, we know that $n - b = a$, and $u = v$. Hence the nodes $y_i$ can be chosen satisfying the demanded conditions by choosing $y_0, \ldots, y_n$ to be the nodes on the $a$-step-path from $w$ to $u = v$, and $y_{n-b}, \ldots, y_n$ to be the nodes on the $b$-path from $u = v$ to $t$.
- If $\text{type} = \text{trans}^k$, we know that $n = a + b + k$, and we can choose $y_0, \ldots, y_n$ to be the nodes on the path from $w$ to $u$, $y_a, \ldots, y_{n-b}$ to denote the $k$-step path from $u$ to $v$ (which exists due to the choice of $u$ and $v$), and $y_{n-b}, \ldots, y_n$ be the nodes on the $b$-step path from $v$ to $t$.
- If $\text{type} = \text{symm}$, then by the above we know that $a = n - b + 1$, and we know that there is an edge $(v, u)$ in $G$. We also know that in this case, $w$ is an $a - 1$-step predecessor of $v$. Hence we can choose the nodes in the following way: Let $y_0, \ldots, y_{n-1} = y_{n-b}$ be chosen as the nodes on the $a - 1$-step path from $w$ to $v$, and let $y_{n-b+1} = y_a, \ldots, y_n$ be the nodes on the $b - 1$-step path from $u$ to $t$. Since $(v, u)$ is an edge in $G$, this gives the edge $(y_{n-b}, y_n)$ which is required since $n - b + 1 = a$.

We now construct the homomorphism $\beta$ for each relevant $i$, let $\beta(x_i) := y_i$. Then by construction, $\beta(x_a) = u$ and $\beta(x_{n-b}) = v$. Hence it remains to prove that $\beta$ is a homomorphism. Since $L$ is a $\text{prev-types-list-line}$, let $L_{\text{strict}}$ be a strict line such that $L$ is the $\text{prev-types-list-closure}$ of $L_{\text{strict}}$. Since $(y_i, y_{i+1})$ is an edge in $G$, and it follows that $\beta: L \rightarrow G$ is a homomorphism. Since every $y_i$ has a $p$-step successor and a $p$-step predecessor in $G$, we know that the subgraph $\{y_0, \ldots, y_n\}$ satisfies the conditions from $\text{prev-types-list}$, and since $L$ is the $\text{prev-list-types-closure}$ of $L_{\text{strict}}$, this implies that for every edge present in $L$, the images of the corresponding vertices are also connected with an edge in $G$, and hence $\beta$ is indeed a homomorphism, finishing the proof of the lemma for the case that $x, y \in \text{prereq}(\hat{\phi})$.

Now assume that $x \notin \text{prereq}(\hat{\phi})$ or $y \notin \text{prereq}(\hat{\phi})$, and therefore $x = y$. Since there obviously is a homomorphism $\alpha$: prereq $(\hat{\phi}) \rightarrow T$ for some $\text{prev-types-list-tree}$ $T$, the clause $\hat{\phi}$ forces every node in a graph containing a $\text{types-list-line}$ of sufficient length to be reflexive. Since due to the induction hypothesis, we know that every model of $\psi$ of sufficient depth contains arbitrary long $\text{prev-types-list-lines}$, this concludes the proof for the remaining case $x = y$. 

Due to Lemma 4.27, we know that the list $\text{types-list}$ maintained by Horn-Classification is sensible, and we are now in a position to prove that the NP-cases claimed by the algorithm are correct as well.

**Lemma 4.28.** Let $\hat{\psi}$ be a universal Horn formula. If Horn-Classification states that $K(\psi)$ has the polynomial-size model property and $K(\psi)$-SAT is in NP, then this is true.
Proof. There are two possibilities for the algorithm to claim the polynomial-size model property, and hence NP-membership. First let us assume that for some $k \geq 2$, both $\text{symm}$ and $\text{trans}^k$ were added to $\text{types-list}$. In this case, due to Lemma 4.27, we know that there is some $p \in \mathbb{N}$ such that $\hat{\psi}$ implies both $\hat{\varphi}^{0-k}_{w \geq p, x \geq p, y \geq p}$ and $\hat{\varphi}^{1-0}_{w \geq p, x \geq p, y \geq p}$. Hence Theorem 4.15 implies both the polynomial-size model property of the logic $K(\hat{\psi})$ and the NP-membership of its satisfiability problem.

The second case in which HORN-CLASSIFICATION claims the NP-result is if it detects a clause $\hat{\varphi}$ which satisfies one of the conditions $1 - 4$. We know by Lemma 4.27, that for each element from $\text{types-list}$, the formula $\hat{\psi}$ implies a formula of the corresponding type. Hence we can assume that there is a natural number $p$, such that the set of vertices in $G$ which have both a $p$-step predecessor and a $p$-step successor satisfy the conditions from $\text{types-list}$. We make a case distinction.

$\hat{\varphi}$ satisfies NP-condition 1. In this case, we know that $\text{conc}(\hat{\varphi}) = (x, y)$ for some $x \neq y \in \text{prereq}(\hat{\varphi})$, and that there is a pair $(\alpha, T) \in \text{types-list} - T_\text{hom}^\text{sym}$ such that there is no directed path connecting $\alpha(x)$ and $\alpha(y)$ in $T$. Note that we can assume $\text{symm} \notin \text{types-list}$, since otherwise every pair of vertices in $T$ would be connected with a directed path. First assume that it is possible to choose $T$ and $\alpha$ in such a way that $\alpha(x) \neq \alpha(y)$. In this case, since $T$ is a tree, we know that $\alpha(x)$ and $\alpha(y)$ have a common predecessor $w$ in $T$. Let $w$ be a common predecessor which is “minimal,” i.e., no node in a lower level than $w$ is a common predecessor.

We construct a $\text{types-list}-$tree $T'$ as a homomorphic image of $T$ via the homomorphism $\beta$ as follows: Let $n$ be the height of the tree $T$, and let $x_0, \ldots, x_n$ be a $\text{types-list}$-line such that every element in $T$ is mapped on its corresponding level in $L$, except the nodes on the path from $w$ to $\alpha(x)$ (excluding $w$) and the successors of $\alpha(x)$. Let $s$ be the level of $w$ in $T$, i.e., let $\alpha(w) = x_s$. Let $w + k$ and $w + l$ be the levels of $\alpha(x)$ and $\alpha(y)$ in $T$, since there is no directed path connecting $\alpha(x)$ and $\alpha(y)$, it follows that $k, l > 0$. Now introduce nodes $y_{s+1}, \ldots$ such that $(x_s, y_{s+1})$ is an edge, and $(y_i, y_{i+1})$ is an edge for every relevant $i$, and map the path from $w$ to $\alpha(x)$ and the successors to $\alpha(x)$ to the “branch” $y_{s+1}, \ldots$ (add as many of these nodes as the “branch” of $T$ requires). Now close the construction under the $\text{types-list}$-condition, and call the tree obtained in this way $T'$. Since $T'$ is the “canonical homomorphic image” of $T$, it is again a $\text{types-list}$-tree, and in $T'$, there is a path of length $k$ from $\beta(w)$ to $\beta(\alpha(x))$ and a path of length $l$ from $\beta(w)$ to $\beta(\alpha(y))$. Let $\hat{\varphi}'$ be the clause with prerequisite graph $T'$ and conclusion edge $(\beta(\alpha(x)), \beta(\alpha(y)))$. Since $\beta \circ \alpha$ is a homomorphism, we know from Proposition 4.4 that $\hat{\varphi}$ implies $\hat{\varphi}'$.

Intuitively, $\hat{\varphi}'$ is the $\text{types-list}$-closure of a clause of the form $\hat{\varphi}^{k-l}_{w \geq p, x \geq q, y \geq n}$, with $q$ chosen according to the height of the tree $T$ and the length of the branch containing $\alpha(x)$ (without loss of generality, we assume that the branch containing $\alpha(x)$ does not contain the deepest node in the tree). We now show that $\hat{\psi} \implies \hat{\varphi}^{k-l}_{w \geq p, x \geq q, y \geq n}$, the complexity result then follows from Corollary 4.14.

Hence, let $G$ be a graph such that $G \models \hat{\psi}$, and let $u$ and $v$ be vertices in $G$ such that $u$ and $v$ have a predecessor $w'$, and $w'$ has a $s + p$-step predecessor, $u$ has a $q + p - k$-step successor, and $v$ has an $n + p - l$-step successor, and there is a $k$-step path from $w'$ to $u$, and an $l$-step path from $w'$ to $v$. Then, since these vertices satisfy the conditions of $\text{types-list}$ by Lemma 4.27, we can homomorphically map the prerequisite graph of $T'$ into $G$ via the homomorphism $\gamma$ such that $(\gamma(\beta(w))) = w'$, $(\gamma(\beta(\alpha(x)))) = u$, and $(\gamma(\beta(\alpha(y)))) = v$. The edges required in order for $\gamma$ to be a homomorphism exist because of the paths of the corresponding lengths connecting $w$, $u$, and $v$, and because all of the relevant nodes in $G$ satisfy the $\text{types-list}$-conditions. Hence, by Proposition 4.3, we know that $(u, v)$ is an edge in $G$, as required to show.
Therefore, assume that it is not possible to choose \((\alpha, T)\) in such a way that \(\alpha(x) \neq \alpha(y)\), i.e., assume that in every \((\alpha, T) \in \text{types-list} - T^\text{hom}_\phi\), \(\alpha(x)\) and \(\alpha(y)\) are connected with a directed path or are identical. Since \(\hat{\varphi}\) satisfies the first NP-condition, we know that there is a pair \((\alpha, T) \in \text{types-list} - T^\text{hom}_\phi\) such that there is no directed path connecting \(\alpha(x)\) and \(\alpha(y)\) in \(T\), and hence \(\alpha(x) = \alpha(y)\), and this node is irreflexive in \(T\). In particular, we know that \text{refl} is not an element of \text{types-list}. We also know that \text{symm} \notin \text{types-list}, since otherwise, every node in the connected graph \(T\) would be connected to any other with a directed path (\(T\) obviously is not the irreflexive singleton, since \(\alpha(x) \neq \alpha(y)\)). Now define \(\beta\) to be \(T\)’s canonical homomorphic mapping to a \(\text{types-list}\)-line \(L = (x_0, \ldots, x_n)\), it then follows that \(\beta(\alpha(x)) = \beta(\alpha(y)) = x_i\) for some \(i\). Since \(x\) and \(y\) are different nodes in \(\text{prereq}(\hat{\varphi})\), we can modify this line as follows: We introduce a new node \(y_i\) which is a “neighbor” to \(x_i\), i.e., a node which is connected to all predecessors and all successors of \(x_i\).

Call this line \(L’\). Note that since \text{types-list} only contains variations of transitivity, there is no condition requiring that \(x_i\) is a reflexive node. Since we connected \(y_i\) to all successors and predecessors of \(x_i\), the line \(L’\) still satisfies all conditions from \text{types-list}, and \(x_i\) and \(y_i\) are not connected with an edge in \(L’\). We now construct a homomorphism \(\gamma: \text{prereq}(\hat{\varphi}) \rightarrow L’\), by defining \(\gamma(z) = \beta(\alpha(z))\) for all \(z \neq y\), and \(\gamma(y) = y_i\). Since the involved nodes are irreflexive and \(\beta \circ \alpha\) is a homomorphism, and \(y_i\) has all edges that \(x_i\) has, we know that \(\gamma\) is a homomorphism. Let \(\hat{\psi}\) be the universal Horn clause with prerequisite graph \(L’\), and conclusion edge \((x_i, y_i)\). The homomorphism \(\gamma\) and Proposition 4.4 show that \(\hat{\varphi}\) implies \(\hat{\psi}\).

Note that the clause \(\hat{\psi}\) requires the following: For every pair of nodes \((x’, y’)\) in a graph \(G\) satisfying the conditions of \text{types-list} which have a common predecessor \(w’\) and a common successor \(z’\) with sufficient height and depth, there is an edge connecting \(x’\) and \(y’\). Since by Lemma 4.27, every graph satisfying \(\psi\) also satisfies the requirements from \text{types-list} for nodes with sufficient depth and height, we can apply \(\hat{\psi}\) to nodes of sufficient depth and height in every such graph.

Let \(G\) be a graph satisfying \(\psi\), and let \(C\) be the set of nodes in \(G\) which have sufficient height and depth to be able to apply the conditions of \(\hat{\psi}\). We show that \(C\) is reflexive, transitive, and symmetric. Note that since \(G\) satisfies \(\psi\), and \(\hat{\varphi}\) is a clause in \(\psi\) implying \(\hat{\psi}\), we know that \(G\) satisfies \(\hat{\psi}\) as well.

We first show that \(C\) is reflexive. Let \(v\) be a node in \(C\). Since \(v\) satisfies the conditions of both \(x_i\) and \(y_i\) in the line \(L’\), and since \(L’ = \text{prereq}(\hat{\varphi})\) and \(G\) satisfies \(\hat{\psi}\), we know that there is an edge \((v, v)\) in \(G\), and hence \(v\) is reflexive.

For symmetry, let there be nodes \(v_1, v_2 \in C\) such that \((v_1, v_2)\) is an edge in \(G\). By the above, we know that both of these nodes are reflexive, and in particular have unbounded height and depth in \(G\). Therefore, we can map the predecessor graph of \(\hat{\psi}\), i.e., the line \(L’\) to these nodes in such a way that for \(j < i\), \(x_i\) is mapped to \(v_1\), \(y_i\) is mapped to \(v_1\), and all \(x_j\) for \(j \geq i\) are mapped to \(v_2\). Since in \(L’\), all edges go from variables with lower indexes to variables

Fig. 9. The homomorphism for the proof that \(C\) is symmetric
with higher indexes, in this way we have constructed a homomorphism from \( \text{prereq}(\hat{\phi}') \) to \( C \), such that the conclusion edge \((x_i, y_i)\) of \( \hat{\phi}' \) is mapped to the pair \((v_2, v_1)\). Hence, by Proposition 4.3, and since \( G \) satisfies \( \hat{\phi}' \), we know that there must be an edge \((v_2, v_1)\) in \( C \), as required.

Finally, we show that \( C \) is transitive. For this, assume that \((s, t)\) and \((t, u)\) are edges in \( C \). Since we already proved symmetry for \( C \), we know that in this case, \((t, s)\) and \((u, t)\) are also edges. We can again construct a homomorphism mapping all elements of \( L' \) to \( t \), except mapping \( x_i \) to \( s \) and \( y_i \) to \( u \). Since the nodes \( s, t, \) and \( u \) are reflexive in \( C \) by the above, and there is no edge \((x_i, y_i)\) or \((y_i, x_i)\) in \( L \), this is a homomorphism. Since \( G \) satisfies \( \phi' \), Proposition 4.3 shows that \((s, u)\) is an edge in \( C \), as claimed.

In particular, since \( C \) is reflexive, symmetric, and transitive, this means that \( C \models \hat{\phi}' \). Since \( C \) is defined as the nodes which have some sufficient depth and height \( p \) in the graph, and \( C \models \hat{\phi}' \), it follows that \( G \models \hat{\phi}' \). Therefore, \( K(\hat{\psi}) \) is a logic extending \( K(\hat{\phi}) \), and the complexity result as well as the polynomial-size model property follow from Corollary 4.14.

\( \hat{\phi} \) satisfies \( \text{NP-condition 2} \). By the prerequisites, since \( \hat{\phi} \) is not satisfied on every \( \text{types-list-tree} \), we can homomorphically map \( \text{prereq}(\hat{\phi}) \) onto a \( \text{types-list-line} \). First assume that \( \text{conc}(\hat{\phi}) \) is empty. Then any graph \( G \) satisfying the conditions of \( \text{types-list} \) which has a line of more than this length does not satisfy the clause \( \hat{\phi} \). Since every graph satisfying \( \hat{\psi} \) also satisfies \( \hat{\phi} \) and the conditions of \( \text{types-list} \) for nodes of sufficient depth and height, every graph satisfying \( \hat{\psi} \) of sufficient depth and height does not satisfy \( \hat{\phi} \), and hence not \( \hat{\psi} \). This implies that graphs satisfying \( \hat{\psi} \) can only be of depth limited by a constant, and hence Lemma 3.8 immediately implies the polynomial-size model property.

Therefore assume that \( \text{conc}(\hat{\phi}) = (x, y) \) for variables \( x, y \), and assume that \( x, y \) are not connected with an undirected path in \( \text{prereq}(\hat{\phi}) \). Since \( x \) and \( y \) lie in different connected components of the underlying directed graph consisting of \( \text{prereq}(\hat{\phi}) \cup \text{conc}(\hat{\phi}) \), and \( \text{prereq}(\hat{\phi}) \) can be homomorphically mapped into a \( \text{types-list-line} \), the “left side” of the implication \( \hat{\phi} \) is satisfied by any pair of vertices \((u, v)\) in a graph satisfying the \( \text{types-list-conditions} \) of sufficient depth and height. Therefore, the subgraph \( C \) containing all nodes with sufficient height and depth forms a universal subgraph, and the NP-result follows by the same reasoning as in Case 1.

\( \hat{\phi} \) satisfies \( \text{NP-condition 3} \). In this case, it follows by the same reasoning as in Lemma 4.27 that any graph satisfying \( \hat{\psi} \) also needs to satisfy the formula \( \hat{\phi}'_{k-0} \) for some \( p \) and some \( k \geq 2 \). Hence the complexity result follows from Corollary 4.14.

\( \hat{\phi} \) satisfies \( \text{NP-condition 4} \). Again, with the proof of Lemma 4.27 it can easily be seen that \( \hat{\psi} \) implies formulas \( \hat{\phi}'_{k-0} \) and \( \hat{\phi}'_{0-k} \) for some \( k \geq 2 \) and some natural number \( p \). Therefore the NP-result follows from Theorem 4.15. \( \square \)
The preceding Lemmas have established that Horn-Classification is well-defined, always comes to a halt, and in each case produces the correct result. Hence, we have proven Theorem 4.17. This theorem and its proof now yield an interesting Corollary:

**Corollary 4.29.** Let $\hat{\psi}$ be a universal Horn formula. If one of the following cases applies:

- $\hat{\psi}$ is satisfied in every strict tree,
- $\hat{\psi}$ is satisfied in every reflexive tree,
- $\hat{\psi}$ is satisfied in every $S$-transitive tree for some $S \subseteq \mathbb{N}$,
- $\hat{\psi}$ is satisfied in every symmetric tree,
- $\hat{\psi}$ is satisfied in every symmetric and reflexive tree,
- $\hat{\psi}$ is satisfied in every $S$-transitive and reflexive tree for some $S \subseteq \mathbb{N}$,

then $K(\hat{\psi})$-SAT is PSPACE-hard and $K(\hat{\psi})$ does not have the polynomial-size model property. In all other cases, $K(\hat{\psi})$ has the polynomial-size model property and $K(\hat{\psi})$-SAT $\in$ NP.

**Proof.** We know by Theorem 4.17 that Horn-Classification correctly determines the complexity of the problem $K(\hat{\psi})$-SAT. By the algorithm, it is obvious that the only arising cases are PSPACE-hard and NP. From the proof of Lemma 4.26, we know that the PSPACE-cases all satisfy the statement of the corollary, and Theorem 2.8 shows that in these cases, we always have PSPACE-hardness.

Finally note that all of our NP-proofs also give the polynomial-size model property, and all PSPACE-hardness proofs also show that this property does not apply. $\square$

### 4.6 Tree-like models for Horn logics

In Section 4.5, we have shown that logics defined by universal Horn formulas have a satisfiability problem which is solvable in NP, or is PSPACE-hard. We now show a tree-like model property for these logics, which we will put to use in the next section by concluding PSPACE-membership for a broad class of these logics. Recall that Corollary 2.6 stated the tree-like model property for the modal logic $K$: Every $K$-satisfiable formula has a tree-like model which also is a $K$-model (which is easy, since every model is a $K$-model). What we want to show is that a logic $K(\hat{\psi})$, where $\hat{\psi}$ is a universal Horn formula satisfied on every reflexive/symmetric/$S$-transitive tree, has the following property: Every modal formula which has a $K(\hat{\psi})$-model also has a $K(\hat{\psi})$-model which is “nearly” the reflexive/symmetric/$S$-transitive closure of a strict tree. In order to prove this result, we first recall from the literature the concept of a bounded morphism, which allows us to prove modal equivalence of models.

**Definition 4.30 ([BdRV01]).** Let $T$ and $M$ be modal models, and let $f: T \rightarrow M$ be a function. Then $f$ is a bounded morphism if the following holds:

(i) For all $w \in T$, $w$ and $f(w)$ satisfy the same propositional variables,
(ii) $f$ is a homomorphism,
(iii) if $(f(u), v')$ is an edge in $M$, then there is some $v \in T$ such that $(u, v)$ is an edge in $T$ and $f(v) = v'$.

The most important feature of bounded morphisms is that they leave the modal properties of the involved models invariant:

**Proposition 4.31 (Proposition 2.14 from [BdRV01]).** Let $T$ and $M$ be modal models, and let $f: T \rightarrow M$ be a bounded morphism. Then for each modal formula $\phi$ and for each $w \in T$, it holds that $T, w \models \phi$ if and only if $M, f(w) \models \phi$. 

With bounded morphisms, we can now prove that our logics have “tree-like” models. We first recall a standard result from the literature about the logic \( K \):

**Proposition 4.32 (Proposition 2.15 from [BdRV01]).** Let \( M, w \models \phi \) such that \( M \) is rooted at \( w \). Then there exists a tree-like model \( T \) and a surjective bounded morphism \( f: T \to M \).

We now generalize this result to universal Horn logics, and “tree-like” models. We already know from Corollary 4.29 that for each universal Horn formula \( \psi \) for which the logic \( K(\psi) \) has a satisfiability problem which cannot be solved in NP, the formula \( \hat{\psi} \) is satisfied on every tree which additionally is closed under reflexivity, \( S \)-transitivity and/or symmetry. We now show that a version of the “converse” is also true: Not only are all of these tree models for the corresponding logics, but for every modal formula satisfiable in such a logic, we can find a model which is “almost” such a tree. The reason for the “almost” is that our Horn formulas usually will not imply a property like \( S \)-transitivity, but only \( S \)-transitivity for nodes at a certain depth in the graph, as shown in Lemma 4.27. We will see in Sections 4.7 and 4.8, that the characterization of the involved models can be used to obtain PSPACE upper bounds for a wide class of logics.

**Theorem 4.33.** 1. Let \( \hat{\psi} \) be a universal Horn formula such that \textsc{Horn-Classification} returns PSPACE-hard on input \( \hat{\psi} \). Let \( \text{types-list} \) be as determined by \textsc{Horn-Classification} on input \( \hat{\psi} \). Since the algorithm determines the logic to be PSPACE-hard, this is well-defined. Then for every modal formula \( \phi \) which is \( K(\hat{\psi}) \)-satisfiable, there exists a \( K(\hat{\psi}) \)-model \( T \) and a world \( w \in T \) such that \( T, w \models \phi \) and there is a strict tree \( T_{\text{strict}} \) such that \( T \) and \( T_{\text{strict}} \) have the same set of vertices, and

\[
\text{edges}(T_{\text{strict}}) \subseteq \text{edges}(T) \subseteq \text{edges}((\text{types-list}(T_{\text{strict}})),
\]

where \( \text{types-list}(T_{\text{strict}}) \) denotes the types-list-closure of \( T_{\text{strict}} \).

2. Let \( \hat{\psi} \) be a universal Horn formula such that \( \hat{\psi} \) is satisfied on every strict line. Then for every modal formula \( \phi \) which is \( K(\hat{\psi}) \)-satisfiable, there exists a \( K(\hat{\psi}) \)-model \( T \) and a world \( w \in T \) such that \( T, w \models \phi \), and \( T \) is \( w \)-canonical.

**Proof.** We prove both claims with nearly the same construction, indicating when differences are required. Since \( \phi \) is \( K(\hat{\psi}) \)-satisfiable, there is a \( K(\hat{\psi}) \)-model \( M \) and a world \( w \in M \) such that \( M, w \models \phi \).

For the first claim, let \( \text{types-list} \) be as determined by \textsc{Horn-Classification} when started on input \( \hat{\psi} \). Since the algorithm does not return NP, we know that \( \text{types-list} \subseteq \{\text{ref1, symm}\} \) or \( \text{types-list} \subseteq \{\text{ref1, trans}^k \mid k \in \mathbb{N}\} \). Since the second NP-condition does not apply, we know that for any clause \( \hat{\varphi} \) in \( \hat{\psi} \) such that there is a homomorphism \( \alpha: \text{prereq}(\hat{\varphi}) \to T \) for some \( \text{types-list}\)-tree \( T \), that \( \text{conc}(\hat{\varphi}) = (x,y) \) for some variables \( x,y \) with \( x = y \) or \( x, y \in \text{prereq}(\hat{\varphi}) \).

For the second claim, we show that for every clause \( \hat{\varphi} \) in \( \psi \), if \( \hat{\varphi} \) is not satisfied in every \( w \)-canonical graph, then \( \text{conc}(\hat{\varphi}) \neq \emptyset \). Obviously, the prerequisite graph of such a clause \( \hat{\varphi} \) can be mapped into some \( w \)-canonical graph, and therefore also into its image \( L \) as a strict line. Since \( \psi \) is satisfied on every strict line, so is \( \varphi \), and hence \( \text{conc}(\varphi) \neq \emptyset \), since otherwise, the clause would be unsatisfied on \( L \), a contradiction.

For both claims, due to Proposition 3.7, we can assume that \( M \) is rooted at \( w \). From Proposition 4.32, we know that there is a model \( T_0 \) which is a strict tree, and a surjective bounded morphism \( f: T_0 \to M \). Let \( T_{\text{strict}} := T_0 \). Let \( w_T \) denote the root of \( T_0 \). The strict tree \( T_0 \) is trivially \( w_T \)-canonical.
We define a sequence of models \((T_n)_{n \in \mathbb{N}}\) such that for each \(n \in \mathbb{N}\), it holds that when we are proving the first claim:

1. vertices \((T_n) = \text{vertices} (T_0)\),
2. edges \((T_{\text{strict}}) \subseteq \text{edges} (T_n) \subseteq \text{edges}(\text{types-list}(T_{\text{strict}}))\),
3. \(f: T_n \to M\) is a homomorphism.

In the case of the second claim, we exchange the second point with “\(T_n\) is \(w_T\)-canonical.”

Note that due to Corollary 3.9, we can assume that the model \(M\) is countable. The proof of Proposition 4.32 from [BdRV01] then constructs a model \(T_0\) which is also countable. Hence assume that \(T_0\) is a countable model. Now let \((e_n)_{n \in \mathbb{N}}\) be a surjective enumeration of vertices \((T_0) \times \text{vertices} (T_0)\), i.e., of all possible edges in the involved trees. The construction of our model is as follows: For \(n = 0\), the model \(T_0\) from above obviously satisfies the conditions, since every bounded morphism is also a homomorphism. For \(n \geq 0\), we make a case distinction:

- If every clause in \(\hat{\psi}\) is satisfied in \(T_n\), then let \(T_{n+1} := T_n\).
- Otherwise, let \(\hat{\varphi}_{n+1}\) be a clause from \(\hat{\psi}\) which is not satisfied in \(T_n\) such that \(\text{conc} (\hat{\varphi}_{n+1}) = (x_{n+1}, y_{n+1})\) for variables \(x_{n+1}, y_{n+1}\), and let \(\alpha_{n+1}: \text{prereq}(\hat{\varphi}_{n+1}) \cup \{x_{n+1}, y_{n+1}\} \to T_n\) be a homomorphism such that \((\alpha_{n+1}(x_{n+1}), \alpha_{n+1}(y_{n+1}))\) is not an edge in \(T_n\). This must exist by Proposition 4.3, since by the above, the case \(\text{conc} (\hat{\varphi}_{n+1}) = \emptyset\) cannot occur. Choose \(\hat{\varphi}_{n+1}, \alpha_{n+1}\) in such a way that the pair \((\alpha_{n+1}(x_{n+1}), \alpha_{n+1}(y_{n+1}))\) has a minimal index in the sequence \((e_n)\). Now let \(T_{n+1}\) be defined with vertex set \(\text{vertices} (T_n)\), edge set \(\text{edges} (T_n) \cup \{(\alpha_{n+1}(x_{n+1}), \alpha_{n+1}(y_{n+1}))\}\), and the same propositional assignments as \(T_n\).

We show that the construction satisfies the requirements 1-3. The first point, which is the same for both claims, holds by definition.

For the first claim, we prove that for each \(n\), it holds that \(\text{edges} (T_{\text{strict}}) \subseteq \text{edges} (T_n) \subseteq \text{edges}(\text{types-list}(T_{\text{strict}}))\). Since by definition, \(\text{edges} (T_n) \subseteq \text{edges} (T_{n+1})\), we know that for all \(n\), it holds that \(\text{edges} (T_{\text{strict}}) = \text{edges} (T_0) \subseteq \text{edges} (T_n)\), and we also know that \(\text{edges} (T_{\text{strict}}) = \text{edges} (T_0) \subseteq \text{edges}(\text{types-list}(T_{\text{strict}}))\).

Hence assume that there is some minimal \(n\) such that \(\text{edges} (T_n) \not\subseteq \text{edges}(\text{types-list}(T_{\text{strict}}))\). Due to the minimality of \(n\), and since the claim holds for \(n = 0\), we know that the edge which is not present in \(\text{edges}(\text{types-list}(T_{\text{strict}}))\) is the edge \((\alpha_n(x_n), \alpha_n(y_n))\). By definition, \(\alpha_n: \text{prereq}(\hat{\varphi}_n) \cup \{x_n, y_n\} \to T_{n-1}\) is a homomorphism. Since due to minimality of \(n\), we know that \(\text{edges} (T_{n-1}) \subseteq \text{edges}(\text{types-list}(T_{\text{strict}}))\), this implies that \(\alpha_n: \text{prereq}(\hat{\varphi}_n) \cup \{x_n, y_n\} \to \text{types-list}(T_{\text{strict}})\) is a homomorphism as well. Since \(\varphi\) is a clause in \(\hat{\psi}\) and \(\text{types-list}(T_{\text{strict}}) \models \hat{\psi}\), we know that \(\text{types-list}(T_{\text{strict}}) \models \hat{\varphi}\), and with Proposition 4.3, we conclude that \((\alpha_n(x_n), \alpha_n(y_n))\) is an edge in \(\text{types-list}(T_{\text{strict}})\), a contradiction. Therefore we know that \(\text{edges} (T_{\text{strict}}) \subseteq \text{edges} (T_n) \subseteq \text{edges}(\text{types-list}(T_{\text{strict}}))\) for all \(n\), and hence we have proven the second point in the case of the first claim.

For the second claim, we need to show that \(T_n\) is a \(w_T\)-canonical graph. For \(i \in \mathbb{N}\), let \(L_i\) denote the nodes in the \(i\)-th level of \(T_{n-1}\), i.e., the set \(\{v \in T_{n-1} | T_{n-1} \models w_T \downarrow^i v\}\). In order to prove that \(T_n\) is \(w_T\)-canonical, since \(T_{n-1}\) is, it suffices to prove that the edge \((\alpha_n(x_n), \alpha_n(y_n))\) in the step from \(T_{n-1}\) to \(T_n\) does not destroy the property of being canonical, i.e., we need to show that \(\alpha_n(x_n) \in L_i\) and \(\alpha_n(y_n) \in L_{i+1}\) for some \(i\). Let \(L\) be the homomorphical image of \(T_{n-1}\) as a strict line via the homomorphism \(\beta\) (since \(T_{n-1}\) is \(w\)-canonical, this exists and is unique). Since \(\alpha_n: \text{prereq}(\hat{\varphi}_n) \cup \{x_n, y_n\} \to T_{n-1}\) is a homomorphism, we know that \(\beta \circ \alpha_n: \text{prereq}(\hat{\varphi}_n) \cup \{x_n, y_n\} \to L\) is a homomorphism as well.
Since \( \hat{\psi} \) is satisfied on every strict line, this is also true for \( \hat{\varphi} \), and hence due to Proposition 4.3, we know that \((\beta \circ \alpha_n(x_n), \beta \circ \alpha_n(y_n))\) is an edge in \( L \). Therefore, \( \alpha_n(x_n) \) is exactly one level above \( \alpha_n(y_n) \) in \( T_{n-1} \), as required.

For both claims, we now show that \( f: T_n \rightarrow M \) is a homomorphism for all \( n \). Again, we prove the fact by induction, and for \( n = 0 \), this holds due to the choice of \( T_0 \), since every bounded morphism is a homomorphism. Therefore, let the claim hold for \( n \), and let \((u,v)\) be an edge in \( T_{n+1} \). If \((u,v)\) is not the edge \((\alpha_{n+1}(x_{n+1}), \alpha_{n+1}(y_{n+1}))\), then we know that \((u,v)\) is an edge in \( T_n \) as well, and since due to induction hypothesis, we know that \( f: T_n \rightarrow M \) is a homomorphism, it follows that \((f(u), f(v))\) is an edge in \( M \). Therefore assume that \( u = \alpha_{n+1}(x_{n+1}) \) and \( v = \alpha_{n+1}(y_{n+1}) \). We need to show that \((f \circ \alpha_{n+1}(x_{n+1}), f \circ \alpha_{n+1}(y_{n+1}))\) is an edge in \( M \). By construction, we know that \( \alpha_{n+1}: \text{prereq}(\hat{\varphi}_{n+1}) \cup \{x_{n+1}, y_{n+1}\} \rightarrow T_n \) is a homomorphism. Since by induction hypothesis, we know that \( f: T_n \rightarrow M \) is a homomorphism, it follows that \( f \circ \alpha_{n+1}: \text{prereq}(\hat{\varphi}_{n+1}) \cup \{x_{n+1}, y_{n+1}\} \rightarrow M \) is a homomorphism as well. Since \( M \) is a \( K(\hat{\psi}) \)-model and \( \hat{\varphi}_{n+1} \) is a clause in \( \psi \), we know that \( M \) satisfies \( \hat{\varphi}_{n+1} \), and thus by Proposition 4.3, we know that \((f \circ \alpha_{n+1}(x_{n+1}), f \circ \alpha_{n+1}(y_{n+1}))\) is an edge in \( M \), as required.

We now construct the desired \( K(\hat{\psi}) \)-model as follows: Define \( T_\infty \) as having \( \text{vertices} (T_\infty) = \{ v \}_{v \in \text{vertices}(T_0)} \) and \( \text{edges} (T_\infty) = \bigcup_{n \in \mathbb{N}} \text{edges}(T_n) \). We show that \( T_\infty \) is a \( K(\hat{\psi}) \)-model, and that \( f: T_\infty \rightarrow M \) is a bounded morphism.

Assume that \( T_\infty \) is not a \( K(\hat{\psi}) \)-model. By construction, we know that in this case, there is no \( n \) such that \( T_n = T_{n+1} \), and therefore each \( T_{n+1} \) has exactly one additional edge in comparison to \( T_n \). Hence we can define a sequence \( (f_n)_{n \in \mathbb{N}} \) such that \( f_n = i \in \mathbb{N} \) iff \( \text{edges}(T_{n+1}) = \text{edges}(T_n) \cup \{ \epsilon_i \} \). Then \( f \) is a sequence of pairwise different natural numbers. Since \( \hat{\psi} \) is not satisfied in \( T_\infty \), there is a clause \( \hat{\varphi} \) from \( \hat{\psi} \) which is not satisfied in \( T_\infty \). For both claims, we therefore know by the above that \( \text{conc}(\hat{\varphi}) = (x, y) \) for variables \( x, y \) with \( x = y \) or \( x, y \in \text{prereq}(\hat{\varphi}) \). By Proposition 4.3, we know that there is a homomorphism \( \alpha: \text{prereq}(\hat{\varphi}) \cup \{x, y\} \rightarrow M \) such that \((\alpha(x), \alpha(y))\) is not an edge in \( T_\infty \). Let \( j \in \mathbb{N} \) and \( (\alpha(x), \alpha(y)) = e_j \). Since in the sequence \( f_n \), no number is repeated, there is some natural number \( n_0 \) such that \( f_n > j \) for all \( n \geq n_0 \). Since \( \text{prereq}(\hat{\varphi}) \) is a finite graph, and \( \text{edges}(T_n) \subseteq \text{edges}(T_{n+1}) \), and \( \text{edges}(T_\infty) = \bigcup_{n \in \mathbb{N}} \text{edges}(T_n) \), we know that there is some \( n_1 \in \mathbb{N} \) such that \( \alpha: \text{prereq}(\hat{\varphi}) \cup \{x, y\} \rightarrow T_{n_1} \) is a homomorphism. Then \( \alpha: \text{prereq}(\hat{\varphi}) \cup \{x, y\} \rightarrow T_n \) is also a homomorphism for all \( n \geq n_1 \), since every edge present in \( T_{n_1} \), is also present in every \( T_n \) for \( n \geq n_1 \). Let \( n := \max(n_0, n_1) \). Then in the step from \( T_n \) to \( T_{n+1} \), the edge \( e_{f_n} \) was added, and by choice of \( n \) we know that \( f_n > j \). Since \( e_j \) is not an edge in \( T_\infty \), we know that \( e_j \) is also not an edge in \( T_{n_1} \). Recall that \( \alpha_{n+1}, \hat{\varphi}_{n+1} \) are chosen in such a way that the edge \((\alpha_{n+1}(x), \alpha_{n+1}(y))\) has minimal index in the sequence \((e_n)_{n \in \mathbb{N}} \), this is a contradiction, since the edge \( e_j \) is an edge with a smaller index than \( e_{f_n} \), and since \( \alpha: \text{prereq}(\hat{\varphi}) \cup \{x, y\} \rightarrow T_{n_1} \) is a homomorphism, \( e_j \) satisfies the conditions of the edge \((\alpha_{n+1}(x), \alpha_{n+1}(y))\) in the construction of \( T_{n_1+1} \).

It remains to show that \( f: T_\infty \rightarrow M \) is a bounded morphism. Property (i) holds by construction, since we do not change propositional assignments, and \( f: T_0 \rightarrow M \) is a bounded morphism. We now show that \( f \) is a homomorphism. Hence let \((u, v)\) be an edge in \( T_\infty \). Since \( T_\infty \) is the union over all \( T_n \), there is some \( n \in \mathbb{N} \) such that \((u, v)\) is an edge in \( T_n \). Since by the above, \( f: T_n \rightarrow M \) is a homomorphism, it follows that \((f(u), f(v))\) is an edge in \( M \), as claimed. For property (iii), let \( u \in T_\infty \) and \( v' \in M \) such that \((f(u), v')\) is an edge in \( M \). Since \( f: T_0 \rightarrow M \) is a bounded morphism, we know that there is some \( v \in T_0 \) such that \((u, v)\) is an edge in \( T_0 \), and \( f(v) = v' \). By construction, \((u, v)\) is also an edge in \( T_\infty \), and therefore \( f: T_\infty \rightarrow M \) is a bounded morphism. By choice of \( f \), and since \( \text{vertices}(T_0) = \text{vertices}(T_\infty) \), \( f \) is also surjective. Hence there is some \( w' \in T_\infty \) such that \( f(w') = w \). Since \( M, w \models \phi \),
it follows from Proposition 4.31 that $T_\infty, w' \models \phi$. By construction, for the first claim it holds that $\text{edges}(T_{\text{strict}}) \subseteq \text{edges}(T_\infty) \subseteq \text{edges}(\text{types-list}(T_{\text{strict}}))$, since this holds for the individual $T_n$ and $T_\infty$ is the union over the edges of all $T_n$, and for the second claim, $T_\infty$ is obviously $w_T$-canonical: Assume that it is not, then there is a node $x \in T_\infty$ and natural numbers $i \neq j$ such that $T_\infty \models w_T \xrightarrow{i} x$ and $T_\infty \models w_T \xrightarrow{j} x$. Since only a finite number of edges is relevant for this path, there exists some $n$ such that $T_n \models w_T \xrightarrow{i} x$ and $T_n \models w_T \xrightarrow{j} x$. This is a contradiction, since $T_n$ is $w_T$-canonical.

Note that we can assume that the node $w'$ is the root of $T_\infty$, since due to Proposition 3.7, we can assume that every node in $T_\infty$ can be reached from $w'$.

4.7 PSPACE upper complexity bounds

In the previous section, we showed that for logics defined by universal Horn formulas, satisfiable formulas are always satisfiable in a tree-like model. Tree-like models are the main argument in many proofs showing PSPACE-membership for satisfiability problems in modal logic. We now show that these models indeed allow us to construct PSPACE algorithms for a wide class of modal logics.

Note that while the construction in the proof of the following theorem has similarities to the constructions by Ladner in [Lad77] or by Halpern and Moses in [HM92], the focus of our result is different. Many proofs of previous PSPACE-algorithms also gave proofs of a variant of some tree-like model property. Our proof relies on this property (which we already proved for our logics in Theorem 4.33), and as a consequence, the verification that the algorithm works correctly with respect to the modal aspect of its task is very easy to verify. The main work of the proof is to prove that the algorithm handles the first-order part of the satisfiability problem correctly, i.e., that the model it constructs is in fact a model satisfying the first-order formula $\psi$ defining the logic. Whereas this is easy for standard classes of frames (checking reflexivity, symmetry, transitivity etc is straightforward), in the general case that we cover here this requires most of the work.

The main feature of the logics that we use here is that of locality: The proof makes extensive use of the fact that in order to verify that the first-order clauses are satisfied it is sufficient to consider local parts of the model constructed by the algorithm. This is the main reason why we believe that this proof does not easily generalize to cases where a variant of transitivity is among the conditions implied by the first-order formula $\psi$.

**Theorem 4.34.** Let $\hat{\psi}$ be a universal Horn formula such that Horn-Classification does not add any element of the form $\text{trans}^k$ to types-list on input $\psi$. Then $K(\psi)$-SAT $\in$ PSPACE.

**Proof.** Since NP $\subseteq$ PSPACE we can assume that $K(\psi)$-SAT $\notin$ NP. Let types-list be as determined by Horn-Classification on input $\hat{\psi}$. From the prerequisites, we know that types-list $\subseteq \{\text{refl, symm}\}$. From Theorem 4.33, we know that for every $K(\hat{\psi})$-satisfiable modal formula $\phi$, there is a model $T$ of $\phi$ such that $T$ is an edge-extension of a strict tree $T_{\text{strict}}$, and every edge present in $T$ which is not an edge of $T_{\text{strict}}$ is a reflexive or a symmetric edge.

By the proof of Lemma 3.8, we can assume that in the tree $T_{\text{strict}}$, every node has at most $|\text{sf}(\phi)|$ successors.

The strategy of the PSPACE-algorithm is as follows: We nondeterministically guess the model $T$ and verify that it is a model of both $\phi$ and of $\hat{\psi}$ by performing a depth-first-search. It is straightforward to see that in this way, we can verify that the modal formula $\phi$ holds in the model. To also check if $\hat{\psi}$ is satisfied requires a bit more effort: Even though we know that $\hat{\psi}$ only adds reflexive or symmetric edges in addition to those present in $T_{\text{strict}}$, we need
to be careful which edges are required and which are not (there may very well be formulas which are not satisfiable on a symmetric and reflexive tree, but are \( K(\psi) \)-satisfiable). The main reason why we can perform these tests is that the properties that we work with have a "local character." To check if an edge is required between some nodes \( u \) and \( v \) in \( T \), we need to know whether there is a clause \( \hat{\phi} \) in \( \bar{T} \) such that \( \hat{\phi} \) can be homomorphically mapped into \( T \) in such a way that the conclusion edge is mapped to the pair \((u, v)\). Since we are working with a tree only extended with reflexive and symmetric edges, we know that a homomorphic image of a connected component of some \( \text{prereq}(\hat{\phi}) \) including \( u \) and \( v \) contains only vertices which are "near" to both \( u \) and \( v \). Therefore we can verify that these clauses are satisfied by procedures looking only locally at the model \( T \).

There are two main obstacles to this approach: For once, the clause \( \hat{\phi} \) that requires \((u, v)\) to be an edge might very well contain more than one connected component except that one containing the conclusion edge (we can ignore the case where there is no conclusion edge, or the vertices from the conclusion edge are unconnected in \( \text{prereq}(\hat{\phi}) \), since if such a clause can be applied, i.e., homomorphically mapped to \( T \), then it can also be homomorphically mapped to the \( \text{types-list-closure} \) of \( T \), and hence to a \( \text{types-list-tree} \), in which case \( \text{HORN-CLASSIFICATION} \) reports \( \text{NP-membership} \), which is a contradiction to our assumption \( K(\psi) \)-\( \text{SAT} \) \( \notin \) \( \text{NP} \). The other obstacle is that although we only need to look at vertices in the "neighborhood" of the current vertex to check that it has all the right edges coming in and out, we need to ensure that all the vertices that we looked at "locally" are consistent, when we revisit a part of the model which is close to a node that we already considered.

The ways to deal with these obstacles is the following: For the first problem, we simply keep a list of connected components of \( \text{prereq}(\hat{\phi}) \)-graphs, and at the beginning of the algorithm, guess for each one if it will appear as a homomorphic image in the tree (which of course later we need to verify). For the second problem, we keep more nodes in storage than just the ones in the neighborhood of the one we are currently visiting, but only a polynomial number.

Strictly speaking, the algorithm does not operate on a model, but on an "annotated model." The annotation of a world is the set of subformulas and negated subformulas of the input formula \( \phi \) which are true at this world, and are required to be true to ensure that the formula \( \phi \) is true at the root-world.

By Proposition 3.7, we can assume that the tree \( T \) has height of at most \( \text{md}(\phi) \). For a node \( v \) in the \( i \)-th level of \( T \), let \( \text{annot}(v) \) denote the set of subformulas and negated subformulas of \( \phi \) which have a modal depth of at most \( \text{md}(\phi) - i \). These are exactly those formulas for which we need to know that they hold at \( v \) in order to verify that the input formula \( \phi \) holds at the root of \( T \).

We now describe the decision procedure, which is a nondeterministic \( \text{PSPACE} \)-algorithm. Let \( S \) be the cardinality of the largest connected component in any of the graphs \( \text{prereq}(\hat{\phi}) \) for clauses \( \hat{\phi} \) of \( \bar{T} \). Note that this number only depends on \( \psi \), and therefore can be regarded as constant. The algorithm as stated in Figure 11 does not work in polynomial space, since it guesses and stores the possibly exponentially-sized model \( T \). We will first show that the algorithm as stated is correct and then prove how it can be implemented using only polynomial space, by only storing a currently relevant subset of the model \( T \).

For the description of the algorithm, we will call a node \( v \in T \) \textit{back-symmetric} if there is an edge \((v, u)\), where \( u \) is the predecessor of \( v \) in \( T_{\text{strict}} \).

When the algorithm guesses the model \( T \), it additionally guesses the set \( \text{annot}(v) \) for every node \( v \) in \( T \), and for each node it guesses if it is back-symmetric and if it is reflexive.

The procedure \( \text{VERIFY-CONSISTENCY}(v) \) performs the following check: For a node \( v \) on the \( i \)-th level of \( T \), \( \text{annot}(i) \) is required to contain all subformulas and negated subformulas
For connected components $C_i$ of all prereq(ϕ), guess if it appears as homomorphic image in $T$
Guess the model $T$
Verify that $φ \in \text{annot}(0)$
current := $w$
while $w$ not marked done do
  Let prereq be the predecessor of current (if current $\neq w$)
  VERIFY-HORN (current)
  if There is $\Diamond \chi \in \text{annot}(\text{current})$ not marked done then
    if current is reflexive and $\chi \in \text{annot}(\text{current})$ then
      Mark $\Diamond \chi$ done in annot(current)
    end if
    if current is back-symmetric and $\chi \in \text{annot}(\text{prereq})$ then
      Mark $\chi$ done
    end if
  end if
  Let next be next unvisited successor of current
  Verify that $\chi \in \text{annot}(\text{next})$
current := next
else
  VERIFY-CONSISTENCY (current)
  if current is reflexive then
    Verify that $\text{annot}(\text{current})$ does not contain $\chi$ and $\neg \Diamond \chi$ for any $\chi$
  end if
  if current is back-symmetric then
    Verify that $\text{annot}(\text{current})$ does not contain $\neg \Diamond \chi$ for $\chi \in \text{annot}(\text{prereq})$
  end if
  Verify that $\text{annot}(\text{prereq})$ does not contain $\neg \Diamond \chi$ for some $\chi \in \text{annot}(\text{current})$
  Mark current as done
  In prereq, mark $\Diamond \chi$ done for all $\chi \in \text{annot}(\text{current})$
current := prereq
end if
end while
Accept

Fig. 11. Algorithm Satisfiability
of $\phi$ which have modal depth of at most $\text{md}(\phi) - i$, and are true at $v$. Hence, $\text{annot}(\text{current})$ must contain exactly one of $\neg \chi$ or $\chi$ for each relevant $\chi$, and additionally, if $\chi_1 \land \chi_2 \in \text{annot}(\text{current})$, then both $\chi_1$ and $\chi_2$ need to be members as well. Similarly, if $\chi_1 \lor \chi_2$ is a member, then at least one of them must be an element of $\text{annot}(\text{current})$.

The procedure $\text{VERIFY-HORN}(v)$ works as follows: If there is a clause $\hat{\varphi}$ in $\hat{\psi}$ with $\text{conc}(\hat{\varphi}) = (x, x)$ for $x \notin \text{prereq}(\hat{\varphi})$ such that all connected components of $\text{prereq}(\hat{\varphi})$ can be mapped homomorphically into $T$, then $\text{VERIFY-HORN}(v)$ ensures that $v$ is reflexive. Note that all other Horn clauses in $\hat{\psi}$ satisfy that $\text{conc}(\hat{\varphi}) = (x, y)$ for some $x, y \in \text{prereq}(\hat{\varphi})$. For these clauses, the procedure considers the subgraph $G_v$ consisting of all nodes of $T$ which can be reached from $v$ in at most $S$ undirected steps (note that a node can be reached in at most $S$ steps in $T$ if and only if it can be reached in at most $S$ steps in $T_{\text{strict}}$). For every connected component $C$ of $\text{prereq}(\hat{\varphi})$, $\text{VERIFY-HORN}(v)$ tests all functions $\alpha : C \rightarrow G_v$. If one of these $\alpha$ is a homomorphism, then $\text{VERIFY-HORN}(v)$ rejects, if the algorithm guessed in the beginning that $C$ cannot be mapped homomorphically into $T$. If there is one clause $\hat{\varphi}$ in $\hat{\psi}$ such that all connected components of $\text{prereq}(\hat{\varphi})$ can be mapped into $T$ (according to the list of these possibilities maintained by the algorithm) and $\text{VERIFY-HORN}(v)$ detected a homomorphism $\alpha : C_{\text{prereq}(\hat{\varphi})} \rightarrow G_v$ (where $C_{\text{prereq}(\hat{\varphi})}$ is the connected component of $\text{prereq}(\hat{\varphi})$ containing the nodes from the conclusion edge of $\hat{\varphi}$) for some $\hat{\varphi}$ with $\text{conc}(\hat{\varphi}) = (x, y)$ such that $v \in \{\alpha(x), \alpha(y)\}$, then $\text{VERIFY-HORN}(v)$ rejects if $(\alpha(x), \alpha(y))$ is not an edge in $G_v$.

We prove that the algorithm is correct. First note that for each connected component $C$ of some $\text{prereq}(\hat{\varphi})$ for a clause $\hat{\varphi}$ in $\hat{\psi}$, if there is a homomorphism $\alpha : C \rightarrow T$, then there is a node $v \in T$ such that $\alpha : C \rightarrow G_v$ is a homomorphism. This holds because all edges in $T$ are already present in the strict tree $T_{\text{strict}}$ or are symmetric or reflexive edges, and the homomorphic image of $C$ under the homomorphism $\alpha$ is a connected component of $T$, and the maximal distance of the nodes in this image is $S$ (recall that this is the maximal cardinality of a connected component in any $\text{prereq}(\hat{\varphi})$). Therefore, we can assume that for each connected component $C$ of some $\text{prereq}(\hat{\varphi})$, if it can be homomorphically mapped into $T$, then $\text{SATISFIABILITY}$ guessed this correctly in the beginning in every accepting run of the algorithm (an incorrect guess would, due to the observation just made, be detected by $\text{VERIFY-HORN}(v)$ for some node $v$).

Now assume that the algorithm accepts. We claim that the model obtained from the annotated model guessed by the algorithm where a variable $x$ is true at a world $v$ if and only if $x \in \text{annot}(v)$ is a model of both the modal formula $\phi$ (at the root-world $w$) and of the Horn formula $\hat{\psi}$. By the checks the algorithm performs, it can easily be verified by induction on the level of the nodes (corresponding to the modal depth of the involved formulas) that for every world $v \in T$, every formula in $\text{annot}(v)$ is satisfied at $v$. Since $\phi \in \text{annot}(w)$, this implies that $T, w \models \phi$. The base case for the induction is clear, since for worlds $v$ in the level $\text{md}(\phi)$, $\text{annot}(v)$ only contains literals, and the algorithm ensures that $\text{annot}(v)$ is propositionally consistent. Since for each subformula $\chi$ of $\phi$ of relevant modal depth, $\text{annot}(v)$ contains exactly one of $\chi$ and $\neg \chi$, the induction hypothesis can be applied in the relevant cases. Note that the algorithm also checks consistency for the cases in which we have reflexive and/or symmetric edges.

It remains to show that $T$ is also a model of the first-order formula $\hat{\psi}$. Assume that this is not the case. Then there exists some clause $\hat{\varphi}$ in $\hat{\psi}$ which is not satisfied in $T$. In particular, this implies that $\text{prereq}(\hat{\varphi})$ can be homomorphically mapped into $T$, and since $\text{edges}(T) \subseteq \text{edges}(\text{types-list}(T))$, this implies that $\text{prereq}(\hat{\varphi})$ can be homomorphically mapped into a $\text{types-list}$-tree. Since we assumed that $\text{HORN-CLASSIFICATION}$ does not return $\text{NP}$ on input $\hat{\psi}$, we know that none of the $\text{NP}$-conditions from $\text{HORN-CLASSIFICATION}$ are
satisfied. Since none of the NP-conditions from HORN-CLASSIFICATION occur, we know that \( \hat{\varphi} \) has conclusion edge \( \text{conc}(\hat{\varphi}) = \{x, y\} \) for variables \( x, y \) where \( x = y \) or \( x, y \in \text{prereq}(\hat{\varphi}) \). First assume that \( x = y \). Since \( \text{prereq}(\hat{\varphi}) \) can be mapped homomorphically into \( T \), the procedure VERIFY-HORN(.) required every node in \( T \) to be reflexive, hence \( \hat{\varphi} \) is satisfied in \( T \). Not assume that \( x, y \in \text{prereq}(\hat{\varphi}) \). Since none of the NP-conditions apply, we know that \( x \) and \( y \) are connected with an undirected path in \( \text{prereq}(\hat{\varphi}) \). In particular, they lie in the same connected component \( C_{\text{prereq}(\hat{\varphi})} \) of \( \text{prereq}(\hat{\varphi}) \). Since \( \hat{\varphi} \) is not satisfied in \( T \), this implies by Proposition 4.3 that there are nodes \( u, v \in T \) such that \( (u, v) \) is not an edge in \( T \), there is no edge \( (u, v) \) in \( T \), and there is a homomorphism \( \alpha: \text{prereq}(\hat{\varphi}) \rightarrow T \) such that \( \alpha(x) = u \), and \( \alpha(y) = v \). Due to the above, we know that \( \alpha: C_{\text{prereq}(\hat{\varphi})} \rightarrow G_v \) is a homomorphism. Therefore, the homomorphism \( \alpha \) was found by VERIFY-HORN(\( u \)) and VERIFY-HORN(\( v \)). Since \( \text{prereq}(\hat{\varphi}) \) can be homomorphically mapped into \( T \), we know that every connected component \( C \) of \( \text{prereq}(\hat{\varphi}) \) can be homomorphically mapped into \( T \), and due to the above, we know that the SATISFIABILITY guessed this correctly in an accepting run of the algorithm. Therefore, the procedure VERIFY-HORN(.) ensured that \( (u, v) \) is an edge in \( T \), a contradiction.

Now assume that \( \phi \) is \( K(\hat{\varphi}) \)-satisfiable. Due to the remarks at the beginning of the proof, we know that in this case, there exists a \( K(\hat{\varphi}) \)-model \( T \) such that \( T \) is an edge-extension of a strict tree, and \( \text{edges}(T) \subseteq \text{edges}(\text{types-list}(T)) \). Therefore the algorithm can guess this model and verify that it satisfies both \( \phi \) and \( \psi \).

It remains to prove that the algorithm can be implemented in nondeterministic polynomial space. The result then follows, since due to a classic result by Savitch [Sav73], NPSPACE = PSPACE. In order to implement the algorithm using only polynomial space, the main change needed compared to the version stated in Figure 11 is how much of the guessed model \( T \) is stored in memory at a given time.

The NPSPACE-implementation does not guess the entire model \( T \) at the start of the algorithm, but guesses each node the moment it is first accessed (either by being created explicitly, or by being explored as an \( S \)-step neighbor of another node by the procedure VERIFY-HORN(.)). It removes the node from memory at a time when it will not be accessed anymore in the remaining execution of the algorithm.

To be precise, the algorithm at all times keeps in its memory the node \text{current} and all of its predecessors, and all nodes which can be reached from these in at most \( S \) steps in the tree \( T_{\text{strict}} \). Since in \( T \), every node has at most \( |\text{sf}(\hat{\varphi})| \) successors in the next level, and \( S \) is a constant, this is a polynomial number of nodes.

We now need to prove that no necessary information is removed from memory, i.e., that no node is first created, then deleted and then accessed again. Note that from the construction of the algorithm, it is obvious that new nodes are visited in a depth-first order.

Therefore assume that this happens for some node \text{node}. Note that any node which gets deleted from memory is not reachable from the root world \( w \) in at most \( S \) steps, and therefore \text{node} is at some level \( i > S \) in the tree \( T_{\text{strict}} \). Let \( v_1 \) be the node for which \text{node} was visited for the first time, i.e., the first node visited such that \text{node} \( \in G_{v_1} \) (recall that \( G_{v_1} \) is the set of nodes which can be reached from \( v_1 \) in at most \( S \) undirected steps in \( T \)). Since \text{node} is deleted from memory and required again later, there is some node \( v_2 \) such that \text{node} cannot be reached from any predecessor of \( v_2 \) in at most \( S \) steps, and a node \( v_3 \) such that \text{node} can be reached from \( v_3 \) in at most \( S \) steps. Let \( a \) be the (uniquely determined) common predecessor of \text{node} and \( v_2 \) with a maximal level in the tree. Then, since \( a \) is a predecessor of \( v_2 \), we know that \text{node} \( \notin G_{a} \). Hence, \text{node} is at least \( S \) levels below \( a \). Since \( T_{\text{strict}} \) is a tree, any node \( t \) such that \text{node} \( \in G_t \) must therefore be a successor of \( a \). In particular, \( v_3 \)
is a successor of \( a \). This is a contradiction, because \textsc{satisfiability} traverses the tree in depth-first-search, and hence does not leave the sub-tree with root \( a \) and re-enters it later.

Therefore we have shown that it is sufficient to keep a polynomial number of nodes in storage, and which nodes to keep can be decided by an easy pattern. Hence it follows that the algorithm can indeed be implemented in nondeterministic polynomial space as required, concluding the proof.

\[ \Box \]

### 4.8 Applications

Theorem 4.17 and 4.34 can be used to classify the complexity of a lot of concrete logics, but they also imply more general results, for which we will give two examples. For once, recall that Ladner proved that all normal modal logics \( KL \) such that \( S4 \) (the logic over all transitive and reflexive frames) is an extension of \( KL \) give rise to a \textsc{Pspace}-hard satisfiability problem.

The following corollary shows that this result is optimal in the sense that every universal Horn logic which is a “proper extension” of \( S4 \) in the way that they imply the conditions of \( S4 \), already gives an \textsc{NP}-solvable satisfiability problem.

**Corollary 4.35.** Let \( \hat{\psi} \) be a universal Horn formula such that \( \hat{\psi} \) implies \( \hat{\varphi}_{\text{refl}} \land \hat{\varphi}_{\text{trans}} \). Then either \( K(\hat{\psi}) = S4 \), or \( K(\hat{\psi}) \) has the polynomial-size model property and \( K(\hat{\psi})\text{-SAT} \in \text{NP} \).

**Proof.** By the prerequisites, we know that \( \hat{\psi} \) is equivalent to \( \hat{\psi} \land \hat{\varphi}_{\text{refl}} \land \hat{\varphi}_{\text{trans}} \). Hence we can, without loss of generality, assume that \( \hat{\varphi}_{\text{refl}} \) and \( \hat{\varphi}_{\text{trans}} \) appear as clauses in \( \hat{\psi} \).

If every clause in \( \hat{\psi} \) is satisfied in every transitive and reflexive tree, then every modal formula \( \phi \) which is satisfiable in a transitive and reflexive tree is \( K(\hat{\psi}) \)-satisfiable. Note that a special case of Theorem 4.33 gives the result that every \( S4 \)-satisfiable formula also is satisfiable in a reflexive and transitive tree. Therefore, every \( S4 \)-satisfiable formula is also \( K(\hat{\psi}) \)-satisfiable, and hence, every \( K(\hat{\psi}) \)-validity is also \( S4 \)-valid. Therefore, \( S4 \) is an extension of \( K(\hat{\psi}) \). Since by Proposition 2.3, \( K(\hat{\psi}) \) is an extension of \( S4 = K(\hat{\varphi}_{\text{refl}} \land \hat{\varphi}_{\text{trans}}) \), this implies that \( K(\hat{\psi}) = S4 \).

Therefore, we can assume that \( \hat{\psi} \) is not satisfied in every reflexive and transitive tree. Now let \( \text{types-list} \) be as determined by \textsc{Horn-Classification} on input \( \hat{\psi} \). Since \( \hat{\varphi}_{\text{refl}} \) and \( \hat{\varphi}_{\text{trans}} \) are clauses in \( \hat{\psi} \), we know that \( \text{refl} \) and \( \text{trans}^2 \) are elements of \( \text{types-list} \). If all elements in \( \text{types-list} \) are of the form \( \text{refl} \) or \( \text{trans}^2 \), then we know (since \( k \)-transitivity is implied by \( 2 \)-transitivity), since \( \hat{\psi} \) is satisfied on every \( \text{types-list}\)-tree, that \( \hat{\psi} \) is satisfied in every reflexive and transitive tree, a contradiction. Therefore, we now that \( \text{symm} \in \text{types-list} \), and hence by construction, \textsc{Horn-Classification} reports \textsc{NP}-membership. Since by Theorem 4.17 the output of the algorithm is correct, we know that \( \text{K}(\hat{\psi})\text{-SAT} \in \text{NP} \), and \( K(\hat{\psi}) \) has the polynomial-size model property, as claimed.

We further can show a \textsc{Pspace} upper bound for all universal Horn logics which are extensions of the logic \( T \), and hence, from Theorem 4.17, conclude that these are all either solvable in \textsc{NP} (and thus \textsc{NP}-complete if they are consistent), or \textsc{Pspace}-complete.

**Corollary 4.36.** Let \( \hat{\psi} \) be a universal Horn formula such that \( \hat{\psi} \) implies \( \hat{\varphi}_{\text{refl}} \). Then \( \text{K}(\hat{\psi})\text{-SAT} \in \text{Pspace} \).

**Proof.** Assume without loss of generality that \textsc{Horn-Classification} determines the logic \( K(\hat{\psi}) \) to have a \textsc{Pspace}-hard satisfiability problem, otherwise the theorem holds trivially, since \( \text{NP} \subseteq \text{Pspace} \). If \( \hat{\psi} \) implies \( \hat{\varphi}_{\text{trans}} \), then the result follows from Corollary 4.35. Hence
assume that this is not the case. Note that in reflexive graphs, $k$-transitivity is equivalent to transitivity. Also note that the conditions requiring a node to have a certain depth or height in a graph are always satisfied in a reflexive graph, because nodes here have infinite depth and height. Therefore, if $\hat{\psi}$ implies a formula of the form $\varphi_{w \geq p, x \geq q, y \geq r}$ for some $2 \leq k$ and some $p, q, r \in \mathbb{N}$, then $\hat{\psi}$ also implies $\varphi_{\text{trans}}$, and due to the above, we can assume that this is not the case. Thus, types-list as determined by HORN-CLASSIFICATION contains no condition of the form $\text{trans}_k$ for any $k \in \mathbb{N}$. The complexity result now follows from Theorem 4.34. □

In a similar way, we can prove that all universal Horn logics which imply a variant of symmetry give rise to a satisfiability problem in PSPACE. A noteworthy difference in the prerequisites of Corollary 4.36 and Corollary 4.37 is that the former requires the reflexivity condition to be implied by the formula $\hat{\psi}$, while the latter only needs a "near-symmetry"-condition as detected by HORN-CLASSIFICATION.

**Corollary 4.37.** Let $\hat{\psi}$ be a universal Horn formula such that HORN-CLASSIFICATION adds $\text{symm}$ to types-list on input $\hat{\psi}$. Then $K(\hat{\psi})$-SAT $\in$ PSPACE. In particular, any universal Horn logic which is an extension of $\mathcal{B}$ has a satisfiability problem solvable in PSPACE.

**Proof.** If $K(\hat{\psi})$-SAT $\in$ NP, the claim trivially holds. Hence, since HORN-CLASSIFICATION is correct due to Theorem 4.17, we can assume that HORN-CLASSIFICATION returns PSPACE-hard, and $\text{symm} \in$ types-list, where types-list is as determined by HORN-CLASSIFICATION. Since HORN-CLASSIFICATION does not report NP, we know from its construction that $\text{trans}_k \notin$ types-list for all $k \in \mathbb{N}$. Hence the complexity result follows from Theorem 4.34. □

## 5 Conclusion and Future Research

We analyzed the complexity of modal logics defined by universal Horn formulas, covering many well-known logics. We showed that the non-trivial satisfiability problems for these logics are either NP-complete or PSPACE-hard, and gave an easy criterion to recognize these cases. Our results directly imply that (unless NP = PSPACE) such a logic has a satisfiability problem in NP if and only if it has the polynomial-size model property. We also demonstrated that a wide class of the considered logics has a satisfiability problem solvable in PSPACE.

Open questions include determining complexity upper bounds for the satisfiability problems for all modal logics defined by universal Horn formulas. We strongly conjecture that all of these are decidable, and consider it possible that all of these problems are in PSPACE. A successful way to establish upper complexity bounds is the guarded fragment [AvBN98,Grä99]. This does not seem to be applicable to our logics, since it cannot be used for transitive logics, and we obtain PSPACE-upper bounds for all of our logics except those involving a variant of transitivity.

The next major open challenges are generalizing our results to formulas not in the Horn class, and allowing arbitrary quantification. Initial results show that even when considering only universal formulas over the frame language, undecidable logics appear. An interesting enrichment of Horn clauses is to allow the equality relation. Preliminary results indicate that Corollary 4.29 holds for this more general case as well.

**Acknowledgments:** We thank the anonymous referees for many hints and suggestions. The second author thanks Thomas Schneider for helpful discussions.
References

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

[BdRV01] P. Blackburn, M. de Rijke, and Y. Venema. *Modal logic*. Cambridge University Press, New York, NY, USA, 2001.

[BG04] B. Bennett and A. Galton. A unifying semantics for time and events. *Artificial Intelligence*, 153(1-2):13–48, 2004.

[BHSS06] M. Bauland, E. Hemaspaandra, H. Schnoor, and I. Schnoor. Generalized modal satisfiability. In *Proceedings of STACS 2006*, pages 500–511, 2006.

[BZ05] C. Baral and Y. Zhang. Knowledge updates: Semantics and complexity issues. *Artificial Intelligence*, 164(1-2):209–243, 2005.

[CDF03] T. Coffey, R. Dojen, and T. Flanagan. On the automated implementation of modal logics used to verify security protocols. In *Proceedings of ISICT 2003*, pages 329–334. Trinity College Dublin, 2003.

[CL94] C. Chen and I. Lin. The computational complexity of the satisfiability of modal horn clauses for modal propositional logics. *Theoretical Computer Science*, 129(1):95–121, 1994.

[FHJ02] U. Frendrup, Hüttel, and J. Jensen. Modal logics for cryptographic processes. In *Proceedings of EXPRESS 02*, 2002.

[Grä99] E. Grädel. Why are modal logics so robustly decidable? *Bulletin of the European Association for Theoretical Computer Science*, 68:90103, 1999.

[Hal95] J. Halpern. The effect of bounding the number of primitive propositions and the depth of nesting on the complexity of modal logic. *Artificial Intelligence*, 75(2):361–372, 1995.

[HM92] J. Halpern and Y. Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial Intelligence*, 54(2):319–379, 1992.

[HMT88] J. Halpern, Y. Moses, and M. Tuttle. A knowledge-based analysis of zero knowledge. In *Proceedings of STOC 1988*, pages 132–147, New York, NY, USA, 1988. ACM Press.

[HR07] J. Halpern and L. Régo. Characterizing the NP-PSPACE gap in the satisfiability problem for modal logic. In *Proceedings of IJCAI 2007*, pages 2306–2311, 2007.

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

[LR86] R. Ladner and J. Reif. The logic of distributed protocols: Preliminary report. In *Proceedings of TARK 1986*, pages 207–222, San Francisco, CA, USA, 1986. Morgan Kaufmann Publishers Inc.

[Ngu05] L. Nguyen. On the complexity of fragments of modal logics. *Advances in Modal Logic - Volume 5*, pages 249–268. King’s College Publications, 2005.

[Sah73] H. Sahlqvist. Completeness and correspondence in the first and second order semantics for modal logic. In *Proceedings of the Third Scandinavian Logic Symposium*, 1973.

[Sav73] W. J. Savitch. Maze recognizing automata and nondeterministic tape complexity. *Journal of Computer and Systems Sciences*, 7:389–403, 1973.

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

[SP06] L. Schröder and D. Pattinson. PSPACE bounds for rank-1 modal logics. In *Proceedings of LICS 2006*, pages 231–242, 2006.

[Spa93] E. Spaan. *Complexity of Modal Logics*. PhD thesis, Department of Mathematics and Computer Science, University of Amsterdam, 1993.