Conservative extensions in modal logic

Silvio Ghilardi	Carsten Lutz	Frank Wolter	Michael Zakharyaschev
U. of Milan	TU Dresden	U. of Liverpool	Birkbeck College
Italy	Germany	UK	UK

Abstract

Every normal modal logic L gives rise to the consequence relation $\varphi \models_L \psi$ which holds iff ψ is true in a world of an L-model whenever φ is true in that world. We consider the following algorithmic problem for L. Given two modal formulas φ_1 and φ_2, decide whether $\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1 in the sense that whenever $\varphi_1 \wedge \varphi_2 \models_L \psi$ and ψ does not contain propositional variables not occurring in φ_1, then $\varphi_1 \models_L \psi$. We first prove that the conservativeness problem is coNExpTime-hard for all modal logics of unbounded width (which have rooted frames with more than N successors of the root, for any $N < \omega$). Then we show that this problem is (i) coNExpTime-complete for **S5** and **K**, (ii) in ExpSpace for **S4** and (iii) ExpSpace-complete for **GL.3** (the logic of finite strict linear orders). The proofs for **S5** and **K** use the fact that these logics have uniform interpolation.

1 Introduction

A theory T_2 is said to be a *conservative extension* of a theory T_1 if any consequence of T_2, which only uses symbols from T_1, is a consequence of T_1 as well. This notion plays an important role in mathematical logic and the foundations of mathematics. For example, the result that the Bernays–Gödel set theory BG (or BGC) is a conservative extension of the Zermelo–Fraenkel set theory ZF (or ZFC) means the relative consistency of BG(C): if ZF(C) is consistent then BG(C) is also consistent.

Rather surprisingly, in modal logic the notion of conservative extension has hardly been investigated. Indeed, modal theories—similarly to first-order theories—have become fundamental tools for representing various domains. For example, in epistemic logic, modal theories represent the (possibly introspective) knowledge of an agent; in temporal logic, theories serve as specifications of concurrent systems; in description logic, theories (called TBoxes) are ontologies used to fix the terminology of an application domain, etc. In all these examples, the notion of a conservative extension can be used to compare different theories and derive important information about their relation to each other: for instance, a temporal specification T_2 can be regarded as a 'safe' refinement of another temporal specification T_1 if, and only if, T_2 is a conservative extension of T_1 (see, e.g., [13]). A description logic ontology T_2 is a 'safe' extension of another description logic ontology T_1 if, and only if, T_2 is a conservative extension of T_1 (see [1, 5]).

One of the main reasons for using modal logic instead of full first-order logic in the applications above is that reasoning in modal logic is often decidable. To employ the notion of conservative extension for modal logics, it is therefore crucial to analyse the algorithmic problem of deciding whether one modal theory is a conservative extension of another modal theory.

In this paper, we investigate the notion of conservative extension for a number of basic modal logics and, in particular, determine the computational complexity of the conservativeness problem for these logics.

In modal logic, the notion of conservative extension depends on the consequence relation we are interested in. Of particular importance are the 'local consequence relation' according to which a formula φ follows from a formula ψ if φ is true *in every world* where ψ is true, and the 'global consequence relation' according to which φ follows from ψ if φ is true *everywhere in a model* whenever ψ is true everywhere in this model (see, e.g., [8]). In this paper, we concentrate on the local consequence relation. Some information about the global consequence relation is provided in the final section.

We begin by showing that deciding non-conservativeness is NExpTime-hard for all modal logics of unbounded width (which have rooted frames with more than N successors of the root, for any $N < \omega$). This result covers almost all standard modal logics, for example, **K**, **S4**, **S5**, and **S4.3**. Thus, deciding conservativeness turns out to be much harder than deciding satisfiability. We also observe that for tabular modal logics (see, e.g., [2, 16]) non-conservativeness is $\mathrm{NP}^{\mathrm{NP}}$-complete, which coincides with the complexity of non-conservativeness in classical propositional logic [9]. The proof of this result and many other proofs in this paper are based on some elementary facts connecting conservativeness with bisimulations.

Next, to warm up, we consider the modal logic **S5** and show that in this case non-conservativeness is NExpTime-complete by proving that one can construct a uniform interpolant of exponential size in exponential time (in the size of a given formula) and using the fact that **S5**-satisfiability is decidable in NP. This proof is based on a general result connecting conservativeness with uniform interpolation (see [10, 15] for a discussion of this variant of interpolation).

A slightly different technique is used to prove that for **K** non-conservativeness is NExpTime-complete. Here we employ a result from [12] according to which there exist uniform interpolants for **K** of (only) exponential size, and then provide a direct algorithm deciding non-conservativeness without computing the uniform interpolant.

After that we consider the non-conservativeness problem for **S4** and establish an ExpSpace upper bound. As this upper bound (probably) does not match the NExpTime lower bound, we leave the exact complexity as an open problem. The logic **S4** does not have uniform interpolation [6], and therefore a 'direct' algorithm had to be found. Similar arguments show that non-conservativeness is decidable in ExpSpace for **K4**, **Grz** and **GL**.

Finally, we prove that conservativeness is ExpSpace-complete for **GL.3**, the logic of finite strict linear orders. Here we again give direct proofs for both lower and upper bounds. Similar proofs show ExpSpace-completeness of conservativeness for **K4.3** and **S4.3**.

2 Preliminaries

We consider the language $\mathcal{ML}$ of propositional unimodal logic with countably many propositional variables $p_1, p_2, \dots$, the Booleans $\wedge$ and $\neg$, and the modal operator $\Box$. Other Boolean operators and the modal diamond $\Diamond$ are defined as usual. Given an $\mathcal{ML}$-formula φ, we denote by $var(\varphi)$ the set of propositional variables occurring in φ.

A (*Kripke*) *frame* $\mathfrak{F} = (W, R)$ is a nonempty set W of points (or worlds) with a binary relation R on it. A (*Kripke*) *model* $\mathfrak{M} = (\mathfrak{F}, \mathfrak{V})$ consists of a frame and a *valuation* $\mathfrak{V}$ giving truth-values to propositional variables in the worlds of W. The *satisfaction relation* '$(\mathfrak{M}, w) \models \varphi$' between *pointed models* $(\mathfrak{M}, w)$ (where $w \in W$) and $\mathcal{ML}$-formulas φ is defined as usual. (If $\mathfrak{M}$ is clear from the context, instead of $(\mathfrak{M}, w) \models \varphi$ we often write $w \models \varphi$.) A formula φ is said to be *valid* in a frame $\mathfrak{F}$ if $(\mathfrak{M}, w) \models \varphi$ holds for every model $\mathfrak{M}$ based on $\mathfrak{F}$ and every point w in it.

Consider now a Kripke complete normal modal logic L (i.e., a subset L of $\mathcal{ML}$ for which there exists a class $\mathcal{F}$ of frames such that L is the set of all formulas that are valid in every $\mathfrak{F} \in \mathcal{F}$). The *local consequence relation* '$\varphi_1 \models_L \varphi_2$' for L is defined as follows: $\varphi_1 \models_L \varphi_2$ holds if, and only if, for every pointed model $(\mathfrak{M}, w)$ based on a frame for L, we have $(\mathfrak{M}, w) \models \varphi_2$ whenever $(\mathfrak{M}, w) \models \varphi_1$.

Given a Kripke complete normal modal logic L and two $\mathcal{ML}$-formulas φ_1 and φ_2, we say that $\varphi_1 \wedge \varphi_2$ is a *conservative extension of* φ_1 in L if, for every $\psi \in \mathcal{ML}$ with $var(\psi) \subseteq var(\varphi_1)$, $\varphi_1 \wedge \varphi_2 \models_L \psi$ implies $\varphi_1 \models_L \psi$.

If $\varphi_1 \wedge \varphi_2$ is *not* a conservative extension of φ_1 in L, then there is a formula ψ with $var(\psi) \subseteq var(\varphi_1)$ such that $\varphi_1 \wedge \psi$ is satisfiable in a model based on a frame for L, while $\varphi_1 \wedge \varphi_2 \wedge \psi$ is not satisfiable in any such model. In this case we call ψ a (non-conservativeness) *witness formula* (or simply a *witness*) for the pair (φ_1, φ_2) in L.

It is worth mentioning that, in propositional classical logic, deciding non-conservativeness is $\mathrm{NP}^{\mathrm{NP}}$-complete (or, which is the same, Σ_2^P-complete). In fact, for propositional formulas φ_1 and φ_2, $\varphi_1 \wedge \varphi_2$ is not a conservative extension of φ_1 iff the formula

$$\exists \mathbf{p}(\varphi_1 \wedge \forall \mathbf{q} \neg(\varphi_1 \wedge \varphi_2))$$

is valid (where $\mathbf{p} = var(\varphi_1)$ and $\mathbf{q} = var(\varphi_2) \setminus var(\varphi_1)$). Clearly, this formula is equivalent to the formula

$$\exists \mathbf{p} \forall \mathbf{q}(\varphi_1 \wedge \neg\varphi_2),$$

and deciding validity of quantified Boolean formulas of this form is known to be $\mathrm{NP}^{\mathrm{NP}}$-complete [9].

The notion of conservative extension turns out to be closely connected with the notion of uniform interpolation. We remind the reader that a modal logic L is said to have *uniform interpolation* if, for every formula φ and every finite set $\mathbf{p}$ of variables, there exists a formula $\exists_L \mathbf{p}.\varphi$ such that

- $var(\exists_L \mathbf{p}.\varphi) \subseteq var(\varphi) \setminus \mathbf{p}$,
- $\varphi \models_L \exists_L \mathbf{p}.\varphi$, and
- $\varphi \models_L \psi$ implies $\exists_L \mathbf{p}.\varphi \models_L \psi$, for every formula ψ with $var(\psi) \cap \mathbf{p} = \emptyset$.

Lemma 1 *If L has uniform interpolation, then $\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1 in L iff $\varphi_1 \models_L \exists_L \mathbf{p}.(\varphi_1 \wedge \varphi_2)$, where $\mathbf{p} = var(\varphi_2) \setminus var(\varphi_1)$.*

Proof. Suppose that $\varphi_1 \wedge \varphi_2$ is not a conservative extension of φ_1. Take a formula ψ with $var(\psi) \subseteq var(\varphi_1)$ such that $\varphi_1 \wedge \varphi_2 \models_L \psi$ and $\varphi_1 \not\models_L \psi$. Then we must have $\exists_L \mathbf{p}.(\varphi_1 \wedge \varphi_2) \models_L \psi$, from which $\varphi_1 \not\models_L \exists_L \mathbf{p}.(\varphi_1 \wedge \varphi_2)$.

Conversely, suppose $\varphi_1 \not\models_L \exists_L \mathbf{p}.(\varphi_1 \wedge \varphi_2)$. But then $\varphi_1 \wedge \varphi_2$ cannot be a conservative extension of φ_1 because $\varphi_1 \wedge \varphi_2 \models_L \exists \mathbf{p}.(\varphi_1 \wedge \varphi_2)$. ❑

This lemma suggests the following procedure for deciding the conservativeness problem for a modal logic L with uniform interpolation: given φ_1 and φ_2, construct $\exists_L \mathbf{p}.(\varphi_1 \wedge \varphi_2)$ with $\mathbf{p} = var(\varphi_2) \setminus var(\varphi_1)$ and then check whether $\varphi_1 \models_L \exists_L \mathbf{p}.(\varphi_1 \wedge \varphi_2)$. Below, we will follow this approach for the modal logics **S5** and **K** which enjoy uniform interpolation [15]. Many standard modal logics, such as **S4**, **K4** or **S4.3**, do not have uniform interpolation, however [6]. In those cases we will provide direct proofs.

An important notion that can be used for analysing conservative extensions (as well as uniform interpolation) is the standard bisimulation between Kripke models [7]. Let $\mathbf{p}$ be a finite set of propositional variables, let $\mathfrak{M}_1$ and $\mathfrak{M}_2$ be models based on Kripke frames (W_1, R_1) and (W_2, R_2), respectively, and let $w_i \in W_i$ for $i = 1, 2$. A $\mathbf{p}$-*bisimulation* between $(\mathfrak{M}_1, w_1)$ and $(\mathfrak{M}_2, w_2)$ is a relation $\sim_{\mathbf{p}}$ between W_1 and W_2 such that the following conditions hold:

- $w_1 \sim_{\mathbf{p}} w_2$,
- if $u_1 \sim_{\mathbf{p}} u_2$, then $(\mathfrak{M}_1, u_1) \models p$ iff $(\mathfrak{M}_2, u_2) \models p$, for every $p \in \mathbf{p}$,
- if $u_1 \sim_{\mathbf{p}} u_2$ and $u_1 R_1 v_1$, then there is some $v_2 \in W_2$ such that $u_2 R_2 v_2$ and $v_1 \sim_{\mathbf{p}} v_2$,
- if $u_1 \sim_{\mathbf{p}} u_2$ and $u_2 R_2 v_2$, then there is some $v_1 \in W_1$ such that $u_1 R_1 v_1$ and $v_1 \sim_{\mathbf{p}} v_2$.

If $(\mathfrak{M}_1, w_1)$ and $(\mathfrak{M}, w_2)$ are $\mathbf{p}$-bisimilar, then we write $(\mathfrak{M}_1, w_1) \leftrightarrows_{\mathbf{p}} (\mathfrak{M}, w_2)$. The main property of bisimilar models is that the second item above follows for all formulas φ with $var(\varphi) \subseteq \mathbf{p}$, that is, if $(\mathfrak{M}_1, w_1) \leftrightarrows_{\mathbf{p}} (\mathfrak{M}, w_2)$ then $(\mathfrak{M}_1, w_1) \models \varphi$ iff $(\mathfrak{M}_2, w_2) \models \varphi$, for all such φ.

We remind the reader that a frame $\mathfrak{F} = (W, R)$ (and a model based on $\mathfrak{F}$) is said to be m-*transitive*, for $m \geq 1$, if whenever $uRx_1R \dots Rx_mRv$ then there exist $k < m$ and points $y_1, \dots, y_k \in W$ such that $uRy_1R \dots Ry_kRv$ (in this sense, standard transitivity is 1-transitivity). A Kripke complete normal modal logic L is called m-*transitive* if the frames validating it are m-transitive.

We will be using the following property of m-transitive models: for every finite set $\mathbf{p}$ of variables and every finite pointed m-transitive model $(\mathfrak{M}, w)$, one can construct a formula $\chi_{\mathbf{p}}(\mathfrak{M}, w)$ containing only variables from $\mathbf{p}$ such that, for every pointed m-transitive model $(\mathfrak{M}', w')$,

$$(\mathfrak{M}', w') \models \chi_{\mathbf{p}}(\mathfrak{M}, w) \quad \text{iff} \quad (\mathfrak{M}, w) \leftrightarrows_{\mathbf{p}} (\mathfrak{M}', w').$$

$\chi_{\mathbf{p}}(\mathfrak{M}, w)$ is called the *characteristic formula* for $(\mathfrak{M}, w)$ and $\mathbf{p}$. Notice that $\chi_{\mathbf{p}}(\mathfrak{M}, w)$ is uniquely determined modulo equivalence in the minimal m-transitive modal logic. Later on in this paper, we will also consider more specialised formulas which satisfy this equivalence only for certain frame classes.

Lemma 2 *For every m-transitive modal logic L with the finite model property, the following conditions are equivalent:*

- *$\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1 in L,*
- *for every finite pointed model $(\mathfrak{M}, w)$ based on a frame for L, if $(\mathfrak{M}, w) \models \varphi_1$ then there exists a finite $var(\varphi_1)$-bisimilar model $(\mathfrak{M}', w')$ based on a frame for L and such that $(\mathfrak{M}', w') \models \varphi_2$.*

Moreover, if $\varphi_1 \wedge \varphi_2$ is not a conservative extension of φ_1 in L, then there exists a finite pointed model $(\mathfrak{M}, w)$ based on a frame for L such that $\chi_{var(\varphi_1)}(\mathfrak{M}, w)$ is a witness for (φ_1, φ_2) in L.

Proof. If $\varphi_1 \wedge \varphi_2$ is not a conservative extension of φ_1, then there is a formula ψ over $\mathbf{p} = var(\varphi_1)$ such that $\varphi_1 \wedge \psi$ is satisfiable but $\varphi_1 \wedge \varphi_2 \wedge \psi$ is unsatisfiable. Take a finite model $(\mathfrak{M}, w)$ based on a frame for L and such that $w \models \varphi_1 \wedge \psi$. But then no pointed model based on a frame for L and $\mathbf{p}$-bisimilar to $(\mathfrak{M}, w)$ can satisfy φ_2.

Conversely suppose that $(\mathfrak{M}, w)$ is a model based on a frame for L such that $(\mathfrak{M}, w) \models \varphi_1$, but $(\mathfrak{M}', w') \models \varphi_2$ does not hold for any finite $var(\varphi_1)$-bisimilar model $(\mathfrak{M}', w')$ based on a frame for L. Then $\varphi_1 \wedge \chi_{var(\varphi_1)}(\mathfrak{M}, w)$ is L-satisfiable, namely, in $(\mathfrak{M}, w)$, but $\varphi_1 \wedge \varphi_2 \wedge \chi_{var(\varphi_1)}(\mathfrak{M}, w)$ is not satisfiable in any finite model based on a frame for L. ❑

As a first application of Lemma 2 we consider the conservativeness problem for tabular logics. A modal logic L is called *tabular* if L is the logic of a finite set of finite frames [2]. It should be clear that every tabular logic is m-transitive for some $m < \omega$.

Theorem 3 *The non-conservativeness problem for each tabular logic L is* $\mathrm{NP}^{\mathrm{NP}}$*-complete.*

Proof. The lower bound follows from $\mathrm{NP}^{\mathrm{NP}}$-hardness of non-conservativeness for propositional logic. The following procedure checks non-conservativeness for a tabular L in $\mathrm{NP}^{\mathrm{NP}}$. Let $\mathcal{F}$ be the finite set of finite rooted frames validating L. Given formulas φ_1 and φ_2, we first guess a pointed model $(\mathfrak{M}, w)$ based on a frame in $\mathcal{F}$. And then we check (i) whether $w \models \varphi_1$ and (ii) whether there does *not* exist a model $(\mathfrak{M}', w') \leftrightarrows_{var(\varphi_1)} (\mathfrak{M}, w)$ based on a frame from $\mathcal{F}$ and satisfying φ_2 at w'. Clearly, (i) can be checked in polynomial time and (ii) by calling an NP-oracle. ❑

3 The NExpTime lower bound

Say that a Kripke complete modal logic L is of *unbounded width* if, for every $N < \omega$, there exist a Kripke frame $\mathfrak{F} = (W, R)$ for L and a point $w \in W$ such that the number of R-successors of w is at least N, or $|\{v \in W \mid wRv\}| \geq N$, to be more precise. Many standard modal logics such as **K**, **K4**, **S4**, **GL**, **S4.3**, **S5** are clearly of unbounded width.

Given a model $\mathfrak{M}$ and a set $\mathbf{q}$ of variables, we call a model $\mathfrak{M}'$ a $\mathbf{q}$-variant of $\mathfrak{M}$ if $\mathfrak{M}'$ can be obtained from $\mathfrak{M}$ by changing the valuation of some variables in $\mathbf{q}$ (but nothing else).

Theorem 4 *Let L be a Kripke complete normal modal logic of unbounded width. Then the conservativeness problem for L is co*NExpTime*-hard.*

Proof. The proof is by reduction of the complement of the well-known NExpTime-complete $2^n \times 2^n$-*bounded tiling problem* (see, e.g., [14]): given $n < \omega$, a finite set $\mathcal{T}$ of tile types and a $t_0 \in \mathcal{T}$, decide whether $\mathcal{T}$ can tile the $2^n \times 2^n$ grid in such a way that t_0 is placed onto $(0,0)$. More precisely, let H and V be the binary relations on $\mathcal{T} \times \mathcal{T}$ such that $(t,t') \in H$ ($(t,t') \in V$) iff the colours of the right (upper) edge of t and the left (bottom) edge of t' coincide. Then $\mathcal{T}$ is said to *tile* the $2^n \times 2^n$ grid if there is a function $\tau : 2^n \times 2^n \to \mathcal{T}$ such that

- if $\tau(i,j) = t$ and $\tau(i+1,j) = t'$ then $(t,t') \in H$, for all $i < 2^n - 1$, $j < 2^n$,
- if $\tau(i,j) = t$ and $\tau(i,j+1) = t'$ then $(t,t') \in V$, for all $i < 2^n$, $j < 2^n - 1$,
- $\tau(0,0) = t_0$.

Given a set $\mathcal{T} = \{t_1, \dots, t_m\}$ of tile types, we are going to construct two formulas φ_1 and φ_2 such that (i) $|\varphi_1|$ and $|\varphi_2|$ are polynomial in m and n, and (ii) $\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1 in L iff $\mathcal{T}$ *cannot* tile the $2^n \times 2^n$ grid in such a way that t_0 is placed onto $(0,0)$.

To construct φ_1 and φ_2, we will use the following propositional variables

- $\mathbf{p} = \{p_1, \dots, p_n\}$ and $\mathbf{q} = \{q_1, \dots, q_n\}$ to represent the points (i,j) of the $2^n \times 2^n$ grid in models by means of the standard binary encoding; for example, $(1,2)$ is represented by a point w of some model iff

  $$w \models p_1 \wedge \neg p_2 \wedge \cdots \wedge \neg p_n \quad \text{and} \quad w \models \neg q_1 \wedge q_2 \wedge \neg q_3 \wedge \cdots \wedge \neg q_n$$

  (we will call a $\mathbf{p}$-*literal* any conjunction $\neg_1 p_1 \wedge \cdots \wedge \neg_n p_n$ where each $\neg_i$ is either $\neg$ or blank),
- $\mathbf{t} = \{t_1, \dots, t_m\}$: $w \models t_i$ will mean that the grid point represented by w is covered by a tile of type t_i,
- the set $\mathbf{A}$ of auxiliary variables $P_1, \dots, P_n$, $Q_1, \dots, Q_n$, $T_1, \dots, T_m$; these variables will occur in φ_2, but not in φ_1.

The formula

$$\varphi_1 \;=\; (\neg p_1 \wedge \cdots \wedge \neg p_n) \wedge (\neg q_1 \wedge \cdots \wedge \neg q_n) \wedge t_0 \wedge \Box \bigvee_{i=1}^{m} t_i.$$

is supposed to say that if $(\mathfrak{M}, w) \models \varphi_1$ then w represents $(0,0)$, which is covered by t_0, and each point of the grid represented by some R-successor of w is covered by at least one tile of a type from $\mathcal{T}$.

We say that a pointed model $(\mathfrak{M}, w)$ *represents a proper tiling of the* $2^n \times 2^n$ *grid by* $\mathcal{T}$ if $(\mathfrak{M}, w) \models \rho_{\mathcal{T},n}$, where $\rho_{\mathcal{T},n}$ is the conjunction of the following formulas:

$$
\begin{aligned}
&\Diamond^+(l_1 \wedge l_2),\\
&\Diamond^+(l_1 \wedge l_2 \wedge t) \to \Box^+\big(l_1 \wedge l_2 \to (t \wedge \neg \bigvee_{t' \neq t} t')\big),\\
&\Diamond^+(l_1 \wedge l_2 \wedge t) \to \Diamond^+ \bigvee_{(t,t') \in H} \big((l_1 + 1) \wedge l_2 \wedge t'\big), \text{ for } l_1 < 2^n - 1\\
&\Diamond^+(l_1 \wedge l_2 \wedge t) \to \Diamond^+ \bigvee_{(t,t') \in V} \big(l_1 \wedge (l_2 + 1) \wedge t'\big), \text{ for } l_2 < 2^n - 1
\end{aligned}
$$

for all possible $\mathbf{p}$-literals l_1, $\mathbf{q}$-literals l_2, and $t, t' \in \mathbf{t}$. Here we use the abbreviations $\Box^+\psi = \psi \wedge \Box\psi$, $\Diamond^+\psi = \psi \vee \Diamond\psi$, and if l_i represents a number $k < 2^n - 1$ then $l_i + 1$ represents $k + 1$.

The formula φ_2 to be constructed below will have the property that, for every model $\mathfrak{M}$ based on a frame (W, R) with $(\mathfrak{M}, w) \models \varphi_1$, the following conditions are equivalent:

1. there is an $\mathbf{A}$-variant $\mathfrak{M}'$ of $\mathfrak{M}$ such that $(\mathfrak{M}', w) \models \varphi_2$,

2. $(\mathfrak{M}, w)$ *does not* represent a proper tiling of the $2^n \times 2^n$ grid by $\mathcal{T}$.

Suppose for the moment that we have managed to construct such a formula φ_2 of length polynomial in m and n. We claim then that $\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1 in L iff $\mathcal{T}$ cannot tile the $2^n \times 2^n$ grid in such a way that t_0 is placed onto $(0,0)$. Indeed, assume first that $\mathcal{T}$ cannot tile the grid in this way, and consider any model $\mathfrak{M}$ based on a frame for L and satisfying φ_1 at its root w. Then $(\mathfrak{M}, w)$ cannot represent a proper tiling of the grid by means of $\mathcal{T}$, and so we can find an $\mathbf{A}$-variant $\mathfrak{M}'$ of $\mathfrak{M}$ such that $(\mathfrak{M}', w) \models \varphi_2$. Clearly, this means that $\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1 in L.

Now suppose that $\mathcal{T}$ can tile the grid. Clearly, we can satisfy $\varphi_1 \wedge \rho_{\mathcal{T},n}$ in a model based on a frame for L. But then $\rho_{\mathcal{T},n}$ is a witness for (φ_1, φ_2) in L because a model $(\mathfrak{M}, w)$ with $(\mathfrak{M}, w) \models \varphi_1 \wedge \varphi_2 \wedge \rho_{\mathcal{T},n}$ would trivially satisfy the former condition above but not the latter.

Now we construct the required formula φ_2. How to ensure that a point and its successors do not represent a tiling properly? There can be three different types of *defects*:

1. two different tiles cover the same point or two different tiles cover two points representing the same pair (i, j) on the grid,

2. there is a point representing (i, j) but here is no point representing $(i+1, j)$, for $i < 2^n - 1$, or no point representing $(i, j+1)$, for $j < 2^n - 1$,

3. colour mismatch.

To encode the *existence* of at least one of these defects we require our auxiliary variables P_i, Q_i, and T_i which will be used to carry, everywhere in the relevant part of the model, the information that there exists a point representing some grid point v covered by some tile t. The formula

$$\bigwedge_{i=1}^{n}(\Diamond^+ P_i \leftrightarrow \Box^+ P_i) \wedge \bigwedge_{i=1}^{n}(\Diamond^+ Q_i \leftrightarrow \Box^+ Q_i) \wedge \bigwedge_{i=1}^{m}(\Diamond^+ T_i \leftrightarrow \Box^+ T_i) \tag{1}$$

says that each of the P_i, Q_i, and T_i has the same truth-value everywhere in the relevant part of the model. Suppose that (1) is true at the root w of our hypothetical model. Then, by making the formula

$$\exists p, q, t \;=\; \Diamond^+\Big(\bigwedge_{i=1}^{n}\big((p_i \leftrightarrow P_i) \wedge (q_i \leftrightarrow Q_i)\big) \wedge \bigwedge_{i=1}^{m}(t_i \leftrightarrow T_i)\Big)$$

true at w, we send—via the P_i, Q_i and T_i—to all worlds accessible from w (and w itself) the information that there is a point v in the model representing some grid point and covered by some tiles. In particular, the formula

$$atP, Q \;=\; \bigwedge_{i=1}^{n}\big((p_i \leftrightarrow P_i) \wedge (q_i \leftrightarrow Q_i)\big)$$

is true at a point u accessible from the root or the root itself iff u and v represent the same grid point, and

$$atP, Q, T \;=\; \bigwedge_{i=1}^{n}\big((p_i \leftrightarrow P_i) \wedge (q_i \leftrightarrow Q_i)\big) \wedge \bigwedge_{i=1}^{m}(t_i \leftrightarrow T_i)$$

is true at u iff u and v represent the same grid point and are covered by the same tiles. Using these formulas we can now express that the model under consideration contains a defect of type 1:

$$\exists p, q, t \wedge \bigvee_{i \neq j}\big(\Diamond^+(atP, Q \wedge t_i) \wedge \Diamond^+(atP, Q \wedge t_j)\big). \tag{2}$$

To describe defects of type 2, we require the formulas

$$\exists p^+, q, t \;=\; \Diamond^+\Big(\neg \bigwedge_{k=1}^{n} p_k \wedge \bigwedge_{k \leq n}\Big(\big(\bigwedge_{i<k} p_i \wedge \neg p_k\big) \to \bigwedge_{i<k} \neg P_i \wedge P_k \wedge \bigwedge_{j=k+1}^{n}(p_j \leftrightarrow P_j)\Big) \wedge \bigwedge_{i=1}^{n}(q_i \leftrightarrow Q_i) \wedge \bigwedge_{i=1}^{m}(t_i \leftrightarrow T_i)\Big)$$

and

$$\exists p, q^+, t \;=\; \Diamond^+\Big(\neg \bigwedge_{k=1}^{n} q_k \wedge \bigwedge_{k\leq n} \Big(\big(\bigwedge_{i<k} q_i \wedge \neg q_k\big) \rightarrow \bigwedge_{i<k} \neg Q_i \wedge Q_k \wedge \bigwedge_{j=k+1}^{n} (q_j \leftrightarrow Q_j)\Big) \wedge \bigwedge_{i=1}^{n} (p_i \leftrightarrow P_i) \wedge \bigwedge_{i=1}^{m} (t_i \leftrightarrow T_i)\Big).$$

The latter, for instance, says that, for some point in the relevant part of the model representing some grid point (k,l) and covered by some tiles t, the variables P_i represent k, the Q_i represent $l+1$, and the T_i represent the same tiles t. (For describing defects of type 2 we do not need the last conjuncts for t_i in these formulas. They will be required for defects of type 3.)

The existence of a defect of type 2 can be guaranteed then by the formula

$$(\exists p^+, q, t \wedge \neg \Diamond^+ atP, Q) \vee (\exists p, q^+, t \wedge \neg \Diamond^+ atP, Q), \tag{3}$$

and the existence of a defect of type 3 can be ensured by the formula

$$\Big(\exists p^+, q, t \wedge \Diamond^+\big(atP, Q \wedge \neg \bigvee_{(t_i,t_j)\in H} (T_i \wedge t_j)\big)\Big) \vee \Big(\exists p, q^+, t \wedge \Diamond^+\big(atP, Q \wedge \neg \bigvee_{(t_i,t_j)\in V} (T_i \wedge t_j)\big)\Big). \tag{4}$$

Finally, we define φ_2 by taking

$$\varphi_2 \;=\; (1) \wedge \big((2) \vee (3) \vee (4)\big).$$

It is easy to see that φ_2 is as required. We leave details to the reader. ❏

4 The upper bound for S5

It is well known that the modal logic **S5** has uniform interpolation. One can easily construct a uniform interpolant $\exists_{\mathbf{S5}}\mathbf{q}.\varphi$ of double exponential size in $|\varphi|$ using the fact that every formula in n variables is equivalent in **S5** to a formula of length $2^{2^{O(n)}}$. It follows from Lemma 1 and the decidability of **S5** in coNP that the non-conservativeness problem for **S5** is decidable in non-deterministic 2ExpTime.

In this section, we improve this bound by showing that **S5** has uniform interpolants of exponential size (which can be constructed in exponential time). Thus, we obtain, by Lemma 1 and Theorem 4, that the conservativeness problem for **S5** is coNExpTime-complete.

Suppose $\varphi(\mathbf{p},\mathbf{q})$ with disjoint $\mathbf{p}$ and $\mathbf{q}$ and $|\mathbf{p}\cup\mathbf{q}| = n$ is given. The $\mathbf{p}$-literal given by a model $\mathfrak{M}$ and a point x in it will be denoted by $l_{\mathfrak{M}}(x)$ or simply $l(x)$ if $\mathfrak{M}$ is understood. Thus,

$$l(x) \;=\; \bigwedge \{p_i \mid (\mathfrak{M},x) \models p_i\} \cup \{\neg p_i \mid (\mathfrak{M},x) \not\models p_i\}.$$

Denote by $M(\varphi)$ the number of occurrences of the modal operator $\Box$ in φ plus one.

A uniform interpolant $\exists_{\mathbf{S5}}\mathbf{q}.\varphi$ of size at most exponential in $|\varphi|$ can be constructed in the following way. First we take the set of all pairwise non-isomorphic rooted **S5**-models over $\mathbf{p}$ and $\mathbf{q}$[1] with at most $M(\varphi)$ worlds and such that no two distinct worlds in a model validate precisely the same variables from $\mathbf{p}$ and $\mathbf{q}$. The total number of such models is not exceeding $2^{n \cdot M(\varphi)}$. Then we partition this set of models into (disjoint) subsets, say $\mathcal{K}_1, \dots, \mathcal{K}_m$, such that all models from the same $\mathcal{K}_i$ validate the same subformulas of φ starting with $\Box$ (recall that we use $\Diamond$ as an abbreviation).

For each $\mathfrak{M} \in \mathcal{K}_i$ based on a frame (W, R) and each point x in $\mathfrak{M}$ with $x \models \varphi$, let

$$\chi_{\mathfrak{M}}(x) \;=\; l(x) \wedge \bigwedge_{w \in W \setminus \{x\}} \Diamond l(w),$$

and let ψ_i be the disjunction of all such $\chi_{\mathfrak{M}}(x)$, for all $\mathfrak{M} \in \mathcal{K}_i$ and x in $\mathfrak{M}$ with $x \models \varphi$. Denote by T_i the set of all $\mathbf{p}$-literals that are not satisfied in any model from $\mathcal{K}_i$, and set

$$\chi_i \;=\; \psi_i \wedge \bigwedge_{l \in T_i} \neg \Diamond l$$

$$\exists_{\mathbf{S5}}\mathbf{q}.\varphi \;=\; \bigvee_{i=1}^{m} \chi_i$$

Clearly, the size of $\exists_{\mathbf{S5}}\mathbf{q}.\varphi$ is at most exponential in the size of φ, and it can be constructed in exponential time. So it remains to prove the following:

Lemma 5 *$\exists_{\mathbf{S5}}\mathbf{q}.\varphi$ is a uniform interpolant for φ in* **S5**.

Proof. To show that $\varphi \models_{\mathbf{S5}} \exists_{\mathbf{S5}}\mathbf{q}.\varphi$, consider an arbitrary rooted **S5**-model $\mathfrak{M}$ based on a frame $(W, W \times W)$ and such that $x \models \varphi$ for some $x \in W$. Let $i \in \{1, \dots, m\}$ be such that $\mathfrak{M}$ validates precisely the same subformulas of φ starting with $\Box$ or $\Diamond$ as the models from $\mathcal{K}_i$. Now observe that, for every point $y \in W$, we can always find a subset $Y \subseteq W$ containing y such that the restriction of $\mathfrak{M}$ to Y is (isomorphic to) some model from $\mathcal{K}_i$ (just pick up one 'witness' satisfying $\neg\psi$ for every subformula $\Box\psi$ of φ such that $y \models \neg\Box\psi$. It follows that $(\mathfrak{M}, x) \models \chi_i$, and so $(\mathfrak{M}, x) \models \exists_{\mathbf{S5}}\mathbf{q}.\varphi$

Suppose now that we have a formula ψ with $var(\psi) \cap \mathbf{q} = \emptyset$. We need to prove that if $\exists_{\mathbf{S5}}\mathbf{q}.\varphi \not\models_{\mathbf{S5}} \psi$ then $\varphi \not\models_{\mathbf{S5}} \psi$. Let $\exists_{\mathbf{S5}}\mathbf{q}.\varphi \not\models_{\mathbf{S5}} \psi$. Take a model $\mathfrak{M} = (\mathfrak{F}, \mathfrak{V})$ with $\mathfrak{F} = (W, W \times W)$ such that $(\mathfrak{M}, w) \models \exists_{\mathbf{S5}}\mathbf{q}.\varphi$ and $(\mathfrak{M}, w) \not\models \psi$ for some $w \in W$. By the definition of $\exists_{\mathbf{S5}}\mathbf{q}.\varphi$, we can find $i \in \{1, \dots, m\}$, $\mathfrak{N} \in \mathcal{K}_i$, and x in $\mathfrak{N}$ such that $(\mathfrak{M}, w) \models \chi_i$ and $(\mathfrak{M}, w) \models \chi_{\mathfrak{N}}(x)$. We know that $(\mathfrak{N}, x) \models \varphi$. Now, with the help of $\mathfrak{M}$, we extend $\mathfrak{N}$ to a model $\mathfrak{K}$ such that $(\mathfrak{K}, x) \models \varphi$ and $(\mathfrak{K}, x) \not\models \psi$.

Let $\mathfrak{N} = (\mathfrak{G}, \mathfrak{U})$ and $\mathfrak{G} = (U, U \times U)$. By the definition of $\chi_{\mathfrak{N}}(x)$, for every $u \in U$ there is $w_u \in W$ such that $l_{\mathfrak{N}}(u) = l_{\mathfrak{M}}(w_u)$. Clearly, we can take $w_x = w$. We can also assume that W is disjoint from U. Now define a new model $\mathfrak{K} = (\mathfrak{H}, \mathfrak{W})$ based on the frame $\mathfrak{H} = (V, V \times V)$, where

$$V \;=\; U \cup (W \setminus \{w_u \mid u \in U\})$$

[1]This means that we restrict valuations in models to the variables in $\mathbf{p}$ and $\mathbf{q}$.

and, for each $u \in U$,

$$\mathfrak{W}(p,u) \;=\; \begin{cases} \mathfrak{U}(p,u), & \text{for } p \in \mathbf{p} \cup \mathbf{q} \\ \mathfrak{V}(p,w_u), & \text{otherwise.} \end{cases}$$

For the remaining points of V the valuation $\mathfrak{W}$ is defined as follows. For each $v \in V \setminus U$ we can find, by the second conjunct of χ_i, a model $\mathfrak{N}_v \in \mathcal{K}_i$ (with a valuation $\mathfrak{U}_v$) and a point z_v in it such that $l_{\mathfrak{N}_v}(z_v) = l_{\mathfrak{M}}(v)$. Then we set, for all such v,

$$\mathfrak{W}(p,v) \;=\; \begin{cases} \mathfrak{U}_v(p,z_v), & \text{for } p \in \mathbf{p} \cup \mathbf{q} \\ \mathfrak{V}(p,v), & \text{otherwise.} \end{cases}$$

We have $(\mathfrak{K},x) \models \varphi$ because, restricted to the variables from $\mathbf{p} \cup \mathbf{q}$, the model $\mathfrak{K}$ consists of $\mathfrak{N} \in \mathcal{K}_i$ together with some points from other models in $\mathcal{K}_i$ validating, by definition, precisely the same subformulas of φ starting with $\Box$. Finally, we have $(\mathfrak{K},x) \not\models \psi$ because $var(\psi) \cap \mathbf{q} = \emptyset$, and, restricted to the variables in $var(\psi)$, the model $\mathfrak{K}$ is isomorphic to $\mathfrak{M}$ with x corresponding to w. ❑

As a consequence we obtain the following:

Theorem 6 *The conservativeness problem for* **S5** *is co*NExpTime*-complete.*

5 The upper bound for K

As in the case of **S5**, it is known [15, 4] that the modal logic **K** has uniform interpolation, with a uniform interpolant for a given formula and set of variables being constructed in an effective way. Moreover, it has been recently shown in [12] that one can construct a uniform interpolant $\exists_{\mathbf{K}}\mathbf{q}.\varphi$ for φ in such a way that the size of $\exists_{\mathbf{K}}\mathbf{q}.\varphi$ is at most exponential in the size $|\varphi|$ of φ — $2^{p(|\varphi|)}$ for a certain polynomial p which does not depend on φ, to be more exact, — and its modal depth does not exceed the modal depth of φ.

Using this result, the fact that the decision problem for **K** is PSpace-complete, and the algorithm from Section 2, one can obtain an algorithm deciding the conservativeness problem for **K** using exponential *space* in the size of the input formulas. In this section we improve this bound by providing a coNExpTime algorithm. Thus, we obtain

Theorem 7 *The conservativeness problem for* **K** *is co*NExpTime*-complete.*

Proof. The coNExpTime lower bound follows from Theorem 4. Here we present a nondeterministic exponential time algorithm for deciding the complement of the conservativeness problem for **K**.

Suppose that we are given formulas $\varphi_1(\mathbf{p})$ and $\varphi_2(\mathbf{p},\mathbf{q})$ with disjoint $\mathbf{p} = \{p_1,\dots,p_n\}$ and $\mathbf{q} = \{q_1,\dots,q_n\}$. Denote by $\mathsf{sub}(\varphi_i)$, $i = 1,2$, the closure under single negation of the set of all subformulas of φ_i. As usual, by a φ_i-*type* t we mean a Boolean-closed subset of $\mathsf{sub}(\varphi_i)$, i.e., Input: formulas $\varphi_1(\mathbf{p})$ and $\varphi_2(\mathbf{p}, \mathbf{q})$.

1. Guess a Kripke model $\mathfrak{M}$ over the variables in $\mathbf{p}$ such that
   - $\mathfrak{M}$ is based on an irreflexive intransitive tree $\mathfrak{F} = (W, R)$,
   - the depth of $\mathfrak{F}$ is $\leq d$,
   - the branching factor of $\mathfrak{F}$ is bounded by $N = |\varphi_2| \cdot 2^{p(|\varphi_1 \wedge \varphi_2|)}$.
2. Check whether $(\mathfrak{M}, r) \models \varphi_1$, where r is the root of $\mathfrak{M}$. If this is not the case, return '$\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1.' Else
3. Label the points x of $\mathfrak{F}$ with subsets $\ell(x)$ of tp_2 by induction starting from the leaves as follows:
   - If $x \in W$ is a leaf of $\mathfrak{F}$, then $\ell(x)$ consists of all types $t \in \mathsf{tp}_2$ such that $p_i \in t$ iff $x \models p_i$, for all $p_i \in \mathbf{p}$, and t contains all formulas of the form $\Box\psi \in \mathsf{sub}(\varphi_2)$.
   - Suppose now that $x \in W$ is not a leaf and all R-successors of x have already been labelled. Consider a type $t \in \mathsf{tp}_2$ and let $\Box\vartheta_1, \ldots, \Box\vartheta_m$ be all the box formulas in t and $\neg\Box\psi_1, \ldots, \neg\Box\psi_k$ be all the negated box formulas in t. Then $t \in \ell(x)$ iff the following conditions hold:
     (a) $p_i \in t$ iff $x \models p_i$, for all $p_i \in \mathbf{p}$,
     (b) for each R-successor y of x, there exists a $t' \in \ell(y)$ with $\{\vartheta_1, \ldots, \vartheta_m\} \subseteq t'$,
     (c) there exist pairwise distinct R-successors $y_1, \ldots, y_k$ of x and types $t_1 \in \ell(y_1), \ldots, t_k \in \ell(y_k)$ such that $\{\neg\psi_i, \vartheta_1, \ldots, \vartheta_m\} \subseteq t_i$, $1 \leq i \leq k$.
4. Check whether there is a $t \in \ell(r)$ such that $\varphi_2 \in t$. If this is the case return '$\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1'; otherwise return 'it is not.'

Figure 1: Deciding non-conservativeness for $\mathbf{K}$.

- $\psi \in t$ iff $\neg\psi \notin t$, for every $\neg\psi \in \mathsf{sub}(\varphi_i)$,
- $\psi \wedge \chi \in t$ iff $\psi \in t$ and $\chi \in t$, for every $\psi \wedge \chi \in \mathsf{sub}(\varphi_i)$.

Denote by tp_i the set of all types for φ_i; clearly, $|\mathsf{tp}_i| \leq 2^{|\varphi_i|}$. Let d be the maximum of the modal depths of φ_1 and φ_2. Now, the algorithm is presented in Fig. 1.

Lemma 8 *The algorithm of Fig. 1 returns '$\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1' iff this is indeed the case.*

Proof. ($\Leftarrow$) Suppose $\varphi_1 \wedge \varphi_2$ is not a conservative extension of φ_1. By Lemma 1, we have $\varphi_1 \not\models_{\mathbf{K}} \exists_{\mathbf{K}}\mathbf{q}.(\varphi_1 \wedge \varphi_2)$. According to [12], there exists a uniform interpolant $\exists_{\mathbf{K}}\mathbf{q}.(\varphi_1 \wedge \varphi_2)$ whose size does not exceed $N' = 2^{p(|\varphi_1 \wedge \varphi_2|)}$ and whose modal depth is $\leq d$. So there is a model $\mathfrak{M}$ based on an irreflexive and intransitive tree $\mathfrak{F} = (W, R)$

of depth $\leq d$ and branching factor $\leq N'$ such that both φ_1 and $\neg\exists_{\mathbf{K}}\mathbf{q}.(\varphi_1 \wedge \varphi_2)$ are true at the root r of $\mathfrak{F}$. Without loss of generality we may assume that, for every $x \in W$, if $x \models \neg\Box\psi_1 \wedge \cdots \wedge \neg\Box\psi_k$ for pairwise distinct $\neg\Box\psi_i \in \mathsf{sub}(\varphi_2)$ then there are k distinct R-successors $y_1, \dots, y_k$ of x such that $y_i \models \neg\psi_i$; if this is not the case, we can duplicate some relevant subtrees, thereby increasing the branching factor to $\leq N$. Thus, we may assume that, restricted to the variables in $\mathbf{p}$, the algorithm above guesses the model $\mathfrak{M}$.

Clearly, in Step 2 of the algorithm, we are in the 'else-case.' Suppose now that there is a $t_0 \in \ell(r)$ with $\varphi_2 \in t_0$, that is, the algorithm returns '$\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1'. Define a function f that maps each $x \in W$ to an element of $\ell(x)$ by induction starting from the root:

- Set $f(r) = t_0$.
- If $f(x)$ is already defined, but $f(\cdot)$ is not defined for the R-successors of x, then do the following. Let the negated box formulas in $f(x)$ be $\neg\Box\psi_1, \dots, \neg\Box\psi_k$ and the box formulas $\Box\vartheta_1, \dots, \Box\vartheta_h$. Since $f(x) \in \ell(x)$, by the definition of ℓ there are distinct R-successors $y_1, \dots, y_k$ of x and types $t_1 \in \ell(y_1), \dots, t_k \in \ell(y_k)$ such that $\{\neg\psi_i, \vartheta_1, \dots, \vartheta_h\} \subseteq t_i$ for $1 \leq i \leq k$. Set $f(y_i) = t_i$ for $1 \leq i \leq k$. For each R-successor y of x such that $y \notin \{y_1, \dots, y_k\}$, there is a $t \in \ell(y)$ such that $\{\vartheta_1, \dots, \vartheta_h\} \subseteq t$. Set $f(y) = t$.

Now define a $\mathbf{q}$-variant $\mathfrak{M}'$ of $\mathfrak{M}$ by taking, for all $x \in W$ and all $p \in \mathbf{q}$,

$$x \models p \quad \text{iff} \quad p \in f(x).$$

We still have $(\mathfrak{M}', r) \models \varphi_1$. Moreover, it can be easily shown by induction that $(\mathfrak{M}', x) \models \bigwedge_{\psi \in f(x)} \psi$ for all $x \in W$. Since $\varphi_2 \in f(r)$, we must have $(\mathfrak{M}', r) \models \varphi_2$. But then $(\mathfrak{M}, r) \models \exists_{\mathbf{K}}\mathbf{q}.(\varphi_1 \wedge \varphi_2)$, which is a contradiction.

($\Rightarrow$) Suppose now that our algorithm returns that $\varphi_1 \wedge \varphi_2$ is not conservative extension of φ_1. Let $\mathfrak{M}$ be the model guessed by the algorithm and based on a tree $\mathfrak{F} = (W, R)$ with root r. Without loss of generality we may assume that, whenever xRy in $\mathfrak{F}$ then there are $|\varphi_2|$-many distinct R-successors z of x with $t^1_{\mathfrak{M}}(z) = t^1_{\mathfrak{M}}(y)$, where $t^1_{\mathfrak{M}}(x)$ the φ_1-*type* of x in $\mathfrak{M}$, that is,

$$t^1_{\mathfrak{M}}(x) \;=\; \{\psi \in \mathsf{sub}(\varphi_1) \mid (\mathfrak{M}, x) \models \psi\}.$$

With each $x \in W$ we associate inductively a formula $\psi(x)$ over $\mathbf{p}$ starting from the leaves of $\mathfrak{F}$:

- if $x \in W$ is a leaf, then
$$\psi(x) \;=\; \Box\bot \wedge \bigwedge_{x \models p_i} p_i \wedge \bigwedge_{x \not\models p_i} \neg p_i,$$
- if $x \in W$ is a non-leaf and $y_1, \dots, y_k$ are its R-successors, then
$$\psi(x) \;=\; \bigwedge_{x \models p_i} p_i \wedge \bigwedge_{x \not\models p_i} \neg p_i \wedge \bigwedge_{1 \leq i \leq k} \Diamond\psi(y_i) \wedge \Box \bigvee_{1 \leq i \leq k} \psi(y_i).$$

We show now that $\varphi_1 \wedge \varphi_2 \models_{\mathbf{K}} \neg\psi$, but $\varphi_1 \not\models_{\mathbf{K}} \neg\psi$. To see the latter, recall first that the algorithm returns '$\varphi_1 \wedge \varphi_2$ is not a conservative extension of φ_1,' and so $\varphi_1 \in t^1_{\mathfrak{M}}(r)$. On the other hand, it follows from the definition of $\psi(r)$ that $(\mathfrak{M}, r) \models \psi(r)$. Therefore, $\varphi_1 \not\models_{\mathbf{K}} \neg\psi$.

To prove the former, we assume to the contrary that the formula $\varphi_1 \wedge \varphi_2 \wedge \psi(r)$ is true at the root r' of some model $\mathfrak{M}'$ based on $\mathfrak{F}' = (W', R')$. By the definition of $\psi(r)$, the relation $\sim$ between W and W' defined by $x \sim x'$ iff $(\mathfrak{M}', x') \models \psi(x)$ is a $\mathbf{p}$-bisimulation between $(\mathfrak{M}, r)$ and $(\mathfrak{M}', r')$.

Let

$$t^2_{\mathfrak{M}'}(x') \;=\; \{\chi \in \mathsf{sub}(\varphi_2) \mid (\mathfrak{M}', x') \models \chi\}.$$

We show by induction starting from the leaves of $\mathfrak{F}'$ that if $x \sim x'$ then $t^2_{\mathfrak{M}'}(x') \in \ell(x)$.

If x' is a leaf then x is also a leaf of $\mathfrak{F}$ and, therefore $x' \models \Box\vartheta$ for all $\Box\vartheta \in \mathsf{sub}(\varphi_2)$. Since, in addition $x' \models \psi(x)$, by the definition of ℓ we must have $t^2_{\mathfrak{M}'}(x') \in \ell(x)$.

Suppose that our claim holds for all R'-successors of x' in $\mathfrak{F}'$. Let $\Box\vartheta_1, \dots, \Box\vartheta_m$ be all box formulas and $\neg\Box\psi_1, \dots, \neg\Box\psi_k$ be all negated box formulas in $t^2_{\mathfrak{M}'}(x')$. We must show that conditions (a)–(c) from Step 3 in the definition of the algorithm hold. The first condition obviously holds because $x' \models \psi(x)$. The second one holds by the induction hypothesis, because for every y with xRy, $y \sim y'$, for some R'-successor y' of x', where we have $\{\vartheta_1, \dots, \vartheta_m\} \subseteq t^2_{\mathfrak{M}'}(y')$. Finally, for every $\neg\Box\psi_i$, $1 \leq i \leq k$, there is an R'-successor y'_i of x' such that $\{\neg\psi_i, \vartheta_1, \dots, \vartheta_m\} \subseteq t^2_{\mathfrak{M}'}(y'_i)$. Let $y_i \sim y'_i$ and xRy_i. By the induction hypothesis, $t^2_{\mathfrak{M}'}(y'_i) \in \ell(y_i)$, and by the condition imposed on the model $\mathfrak{M}$ above, we can always choose pairwise distinct points y_i for $1 \leq i \leq k$.

It follows that $\varphi_2 \in t^2_{\mathfrak{M}'}(r') \in \ell(r)$, and so our algorithm must return '$\varphi \wedge \varphi_2$ is a conservative extension of φ_1', contrary to the original assumption. ❑

To complete the proof of Theorem 7, we note that the algorithm above runs in exponential time: the guessed model $\mathfrak{M}$ is at most exponentially large and checking whether φ_1 is true in r can be done in exponential time. It remains to consider the $\ell(\cdot)$ labelling procedure. We have to label $|W|$ points—i.e., at most exponentially many. For each point x, we have to check for $|\mathsf{tp}_2|$ (exponentially) many types whether or not they should be included in $\ell(x)$. Condition (a) can be checked in polynomial time. Condition (b) can be checked in exponential time, since there are at most exponentially many successors in $\mathfrak{M}$. For condition (c) we have to consider all k-tuples of pairs (y, t) with y a successor of x and $t \in \ell(y)$. It is clear that there are at most exponentially many such tuples. ❑

6 The upper bound for S4

In this section we present an algorithm deciding the conservativeness problem for **S4** in ExpSpace in the size of the input formulas.

Before proceeding to the technical details, we remind the reader that Kripke models for **S4** are based on quasi-orders $\mathfrak{F} = (W, R)$, that is, R is transitive and reflexive. A set $C \subseteq W$ is called a *cluster* in $\mathfrak{F}$ if $C = \{y \in W \mid xRy \;\&\; yRx\}$ for some $x \in W$; in this case we also say that C is the cluster *generated by* x and denote it by $C(x)$.

Recall that every rooted model for **S4** is a p-morphic image of a model based on a *tree of clusters*, that is, a rooted quasi-order (W, R) such that, for all $x, y, z \in W$, if xRz and yRz, then either xRy or yRx or $C(x) = C(y)$. Without loss of generality we assume in this section that all our models are based on *finite trees of clusters*. Given a quasi-order $\mathfrak{F} = (W, R)$, we say that a cluster $C(x)$, for $x \in W$, is an *immediate strict predecessor* of a cluster $C(y)$ if xRy, $C(x) \neq C(y)$ and whenever $xRzRy$ then either $C(z) = C(x)$ or $C(z) = C(y)$. By the *depth* of $\mathfrak{F}$ we understand the length n of the longest sequence $C(x_1), \dots, C(x_n)$ of clusters in $\mathfrak{F}$ such that $C(x_i)$ is an immediate strict predecessor of $C(x_{i+1})$. A point y is a *strict successor* of a point x iff xRy and $C(x) \neq C(y)$. The *branching factor* of $\mathfrak{F}$ is the maximal number of immediate strict successor clusters of a cluster in $\mathfrak{F}$.

Suppose that we are given two formulas φ_1 and φ_2. The central role in our algorithm will be played by the following notion of a realisable triple for φ_1, φ_2. Consider a triple $\mathsf{t} = (t, \Gamma, \Delta)$ where t is a φ_1-type and Γ, Δ are sets of $\varphi_1 \wedge \varphi_2$-types. We call t *realisable* if there exists a pointed model $(\mathfrak{M}, x)$ based on a tree of clusters with root x such that

- $t = t^1_{\mathfrak{M}}(x)$ (where, as before, $t^1_{\mathfrak{M}}(x) = \{\psi \in \mathsf{sub}(\varphi_1) \mid (\mathfrak{M}, x) \models \psi\}$),
- Γ is the set of all $\varphi_1 \wedge \varphi_2$-types s such that $\bigwedge_{\sigma \in s} \sigma \wedge \chi_{var(\varphi_1)}(\mathfrak{M}, x)$ is satisfiable (in some model based on a tree of clusters),
- Δ is the set of all $\varphi_1 \wedge \varphi_2$-types s for which there exists a point y in $\mathfrak{M}$ such that $\bigwedge_{\sigma \in s} \sigma \wedge \chi_{var(\varphi_1)}(\mathfrak{M}, y)$ is satisfiable. (In what follows we will often not distinguish between the type t and the conjunction $\bigwedge_{\sigma \in t} \sigma$, and write, for example, $t \wedge \chi$ instead of $\bigwedge_{\sigma \in t} \sigma \wedge \chi$.)

In this case we say that $\mathsf{t} = (t, \Gamma, \Delta)$ is *realised* by $(\mathfrak{M}, x)$. Observe that if (t, Γ, Δ) is realisable, then $\Gamma \subseteq \Delta$ and, as follows from the main property of characteristic formulas and bisimulations, $t \subseteq s$ for every $s \in \Gamma$. The meaning of realisable triples will become clear from the following lemma.

Lemma 9 *The following two conditions are equivalent for any formulas φ_1 and φ_2:*

(1) *$\varphi_1 \wedge \varphi_2$ is not conservative extension of φ_1,*

(2) *there exists a realisable triple $\mathsf{t} = (t, \Gamma, \Delta)$ (for φ_1, φ_1) such that $\varphi_1 \in t$ but $\varphi_1 \wedge \varphi_2 \notin s$ for any $s \in \Gamma$.*

Moreover, for any finite model $(\mathfrak{M}, x)$ based on a tree of clusters, $\chi_{var(\varphi_1)}(\mathfrak{M}, x)$ is a witness for (φ_1, φ_2) iff the triple t realised by $(\mathfrak{M}, x)$ satisfies condition (2).

Proof. (1) $\Rightarrow$ (2) By Lemma 2, there is a finite pointed model $(\mathfrak{M}, x)$ such that $\chi_{var(\varphi_1)}(\mathfrak{M}, x)$ is a witness for (φ_1, φ_2). Define $\mathsf{t} = (t, \Gamma, \Delta)$ by taking

- $t = t^1_{\mathfrak{M}}(x)$,
- $\Gamma = \{t^{1,2}_{\mathfrak{M}'}(x') \mid (\mathfrak{M}', x') \leftrightarrows_{var(\varphi_1)} (\mathfrak{M}, x)\}$, where $t^{1,2}_{\mathfrak{M}'}(x')$ is the $\varphi_1 \wedge \varphi_2$-type of x' in $\mathfrak{M}'$,

- $\Delta = \{t^{1,2}_{\mathfrak{M}'}(y') \mid (\mathfrak{M}', y') \leftrightarrows_{var(\varphi_1)} (\mathfrak{M}, y) \text{ for some point } y \text{ in } \mathfrak{M}\}$.

It follows from the main property of characteristic formulas that t is a realisable triple. It should be clear that $\varphi_1 \in t$ but $\varphi_1 \wedge \varphi_2 \notin s$ for any $s \in \Gamma$.

(2) $\Rightarrow$ (1) Let $(\mathfrak{M}, x)$ be a pointed model realising t. It is readily seen that $\chi_{var(\varphi_1)}(\mathfrak{M}, x)$ is a witness for (φ_1, φ_2). ❑

We first use the notion of realisable triple to show that whenever $\varphi_1 \wedge \varphi_2$ is not a conservative extension of φ_1, then there exists a witness $\chi_{var(\varphi_1)}(\mathfrak{M}, x)$ of a certain bounded size. As a first step, we show the following

Lemma 10 *Let $\mathfrak{M}$ be a model based on a tree of clusters $\mathfrak{F} = (W, R)$ with root x, and let $(\mathfrak{M}, x)$ realise a triple $\mathsf{t}_x = (t_x, \Gamma_x, \Delta_x)$ for given φ_1, φ_2. Suppose also that $(\mathfrak{N}, z)$ is a model based on a tree of clusters with root z, and for some $y \in W$ we have:*

- *$(\mathfrak{M}, y)$ realises a triple $\mathsf{t} = (t, \Gamma, \Delta)$ for φ_1, φ_2,*
- *$(\mathfrak{N}, z)$ realises a triple $\mathsf{t}' = (t, \Gamma', \Delta')$ for φ_1, φ_2,*
- *$\Gamma \supseteq \Gamma'$ and $\Delta \supseteq \Delta'$.*

Denote by $(\mathfrak{M}', x)$ the result of replacing the model generated by y in $\mathfrak{M}$ with $\mathfrak{N}$, and let $(t_x, \Gamma'_x, \Delta'_x)$ be the triple realised by $(\mathfrak{M}', x)$. Then $t_x = t'_x$, $\Gamma_x \supseteq \Gamma'_x$, and $\Delta_x \supseteq \Delta'_x$.

Proof. That $t_x = t'_x$ is easily proved by induction on the construction of formulas in $\mathsf{sub}(\varphi_1)$.

Suppose now that $s \in \Gamma'_x$. Then we have a model $\mathfrak{K}'$ based on a tree of clusters with root r such that s is realised at r and there is a $var(\varphi_1)$-bisimulation $\sim$ between $(\mathfrak{M}', x)$ and $(\mathfrak{K}', r)$. Without loss of generality we may assume that actually $\sim$ is a function from the set of worlds in $\mathfrak{K}'$ onto the set of worlds in $\mathfrak{M}'$. Consider a maximal connected submodel of $\mathfrak{K}'$ all points of which are $\sim$-related to some points from $\mathfrak{N}$. Then this submodel is generated by a point u; we denote it by $\mathfrak{K}'_u$. Let s' be the $\varphi_1 \wedge \varphi_2$-type realised by $(\mathfrak{K}'_u, u)$. Then $s' \in \Delta'$, and so $s' \in \Delta$. This means that, for some point v in $\mathfrak{M}_y$, we have a model $\mathfrak{K}_{v'}$ based on a tree of clusters with root v' which is $var(\varphi_1)$-bisimilar to $(\mathfrak{M}_v, v)$ and satisfies s' at v'. Let $\sim_{v'}$ be the corresponding bisimulation. Replace the submodel $(\mathfrak{K}'_u, u)$ of $\mathfrak{K}'$ with $(\mathfrak{K}_{v'}, v')$. Since the same type s' is satisfied by $(\mathfrak{K}'_u, u)$ and $(\mathfrak{K}_{v'}, v')$, the resulting model still satisfies s at r. We do the same for all such maximal connected submodels $(\mathfrak{K}'_u, u)$ of $\mathfrak{K}'$, and denote the resulting model by $(\mathfrak{K}, r)$. Define a relation $\approx$ between $(\mathfrak{M}, x)$ and $(\mathfrak{K}, r)$ as follows. For each w in $\mathfrak{M}$ that is not R-accessible from y, we set $w \approx w'$ iff $w \sim w'$ (here we use the fact that $\sim$ is a function from $\mathfrak{K}'$ onto $\mathfrak{M}'$). And for w in $\mathfrak{M}_y$ we set $w \approx w'$ iff $w \sim_{v'} w'$ for some v' in $\mathfrak{K}$. We leave it to the reader to check that $\approx$ is a $var(\varphi_1)$-bisimulation between $(\mathfrak{M}, x)$ and $(\mathfrak{K}, r)$. It follows that $s \in \Gamma_x$, from which $\Gamma_x \supseteq \Gamma'_x$.

The inclusion $\Delta_x \supseteq \Delta'_x$ is proved analogously. ❑

With every realisable triple $\mathsf{t} = (t, \Gamma, \Delta)$ we associate the set

$$\Phi_{\mathsf{t}} = \{\{\Box\psi_1, \dots, \Box\psi_k\} \subseteq \mathsf{sub}(\varphi_1 \wedge \varphi_2) \mid \forall s \in \Gamma\ \{\Box\psi_1, \dots, \Box\psi_k\} \not\subseteq s\}.$$

We are in a position now to prove the following:

Lemma 11 *For every realisable triple* $\mathsf{t} = (t, \Gamma, \Delta)$ *for* φ_1, φ_2, *there is a realisable triple* $\mathsf{t}' = (t, \Gamma', \Delta')$ *such that* $\Gamma \supseteq \Gamma'$, $\Delta \supseteq \Delta'$, *and* t' *can be realised in a model* $\mathfrak{M}'$ *based on a tree of clusters* $\mathfrak{F}'$ *such that*

- *each cluster in* $\mathfrak{F}'$ *contains at most* $2^{|\varphi_1|}$ *points,*
- *the branching factor of* $\mathfrak{F}'$ *is bounded by* $2^{|\varphi_1 \wedge \varphi_2|}$,
- *the depth of* $\mathfrak{F}'$ *is bounded by* $1 + 2^{|\varphi_1 \wedge \varphi_2|}$.

Moreover, for any two points x, y *such that* y *is a strict successor of* x *in* $\mathfrak{F}'$ *and* x *is not in the root cluster of* $\mathfrak{F}'$, $\Phi_{\mathsf{t}_y} \subsetneq \Phi_{\mathsf{t}_x}$, *for the triples* t_x *and* t_y *realised by* $(\mathfrak{M}', x)$ *and* $(\mathfrak{M}', y)$, *respectively.*

Proof. Suppose that a triple $\mathsf{t} = (t, \Gamma, \Delta)$ for φ_1, φ_2 is realised in a model $(\mathfrak{M}, x)$ based on a finite tree of clusters $\mathfrak{F} = (W, R)$ with root x. The upper bound on the cardinality of clusters follows from the simple fact that if two points y and y' validate the same variables from φ_1 then we can omit one of these points and the resulting models will be $var(\varphi_1)$-bisimilar to the original one.

Consider now some $y \in W$ and denote by $\mathsf{t}_y = (t_y, \Gamma_y, \Delta_y)$ the triple for φ_1, φ_2 realised by $(\mathfrak{M}, y)$. For every $\varphi_1 \wedge \varphi_2$-type $s \notin \Gamma_y$ we take the set $\{\Box\psi_1, \dots, \Box\psi_k\}$ of all box formulas in s and choose a maximal (with respect R) strict successor z of y such that there does not exist a type $s' \in \Gamma_z$ with $\{\Box\psi_1, \dots, \Box\psi_k\} \subseteq s'$, if such a strict R-successor exists. Observe that (since $t_y \subseteq s$ for every $s \in \Gamma_y$) for each $\neg\Box\psi \in t_y$ we have also chosen a maximal strict R-successor z of y with $(\mathfrak{M}, z) \models \neg\Box\psi$, if such a successor exists.

Now remove from the submodel $\mathfrak{M}_y$ of $\mathfrak{M}$ generated by y all clusters $C(u)$ such that $C(u) \neq C(y)$, yRu and for no chosen point z do we have zRu. Denote the resulting model by $\mathfrak{N}$. Our aim is to show that $(\mathfrak{N}, y)$ satisfies the conditions of Lemma 10, and so we can replace $\mathfrak{M}_y$ in $\mathfrak{M}$ with $\mathfrak{N}$ where the number of immediate strict successors of $C(y)$ does not exceed $2^{|\varphi_1 \wedge \varphi_2|}$.

Denote by $\mathsf{t}' = (t', \Gamma', \Delta')$ the triple realised by $(\mathfrak{N}, y)$. Since every $\neg\Box\psi \in t_y$ has a witness in $\mathfrak{N}$, we clearly have $t_y = t'$. To prove that $\Gamma' \subseteq \Gamma_y$, suppose otherwise. Then we have some $s \in \Gamma'$ such that $s \notin \Gamma_y$. Two cases are possible now. *Case 1*: there is a strict successor z of y in $\mathfrak{M}_y$ such that there does not exist a type $s' \in \Gamma_z$ with $\{\Box\psi_1, \dots, \Box\psi_k\} \subseteq s'$ (where $\{\Box\psi_1, \dots, \Box\psi_k\}$ are all box formulas in s). Then a (possibly different) point with this property was chosen for $\mathfrak{N}$. From this one can easily derive a contradiction with s being in Γ'. *Case 2*: there is no such z in $\mathfrak{M}_y$. Let $C(y_1), \dots, C(y_m)$ be all immediate strict successors of $C(y)$ which do not belong to $\mathfrak{N}$. Denote by $\mathfrak{M}_i$ the submodel of $\mathfrak{M}_y$ generated by y_i. For each y_i there exists a pointed model $(\mathfrak{M}'_i, y'_i)$ which is $var(\varphi_1)$-bisimilar to $(\mathfrak{M}_i, y_i)$ such that

Input: formulas φ_1, φ_2.

1. Choose, non-deterministically, a triple (t, Γ, Δ), where t is a φ_1-type containing φ_1 and Γ, Δ are sets of $\varphi_1 \wedge \varphi_2$-types such that $\varphi_1 \wedge \varphi_2 \notin s$ for any $s \in \Gamma$.

2. Return '$\varphi_1 \wedge \varphi_2$ is not a conservative extension of φ_1' iff $\mathsf{realise}_0(t, \Gamma, \Delta)$ returns 'true.'

Figure 2: Deciding non-conservativeness for **S4**.

$(\mathfrak{M}'_i, y'_i) \models \Box\psi_i$ for every $\Box\psi \in s$. Let $\mathfrak{K}$ be a model based on a tree of clusters with root v which is $var(\varphi_1)$-bisimilar to $(\mathfrak{N}, y)$ and satisfies s at v. Add the models $\mathfrak{M}'_i$ to $\mathfrak{K}$ as immediate strict successors of the points which are $var(\varphi_1)$-bisimilar to y and denote the resulting model by $\mathfrak{K}'$. Clearly, $(\mathfrak{K}', v)$ is $var(\varphi_1)$-bisimilar to $(\mathfrak{M}_y, y)$. On the other hand, s is still satisfied by $(\mathfrak{K}', v)$, contrary to $s \notin \Gamma_y$.

Finally, the inclusion $\Delta' \subseteq \Delta_y$ follows from $\Gamma' \subseteq \Gamma_y$ and the fact that if a point from a non-root cluster is chosen for $\mathfrak{N}$ then all of its successors belong to $\mathfrak{N}$ as well.

By recursively performing this operation whenever possible for some y, we obtain a model with the properties required. ❑

We are now in a position to present a NExpSpace algorithm deciding the non-conservativeness problem for **S4**. The ExpSpace upper bound is then obtained from the fact that NExpSpace = ExpSpace. To formulate the algorithm, we require the following definition. Let $\mathcal{R}$ be some set of realisable triples. We say that a triple $\mathtt{t}$ is *obtained in one step from* $\mathcal{R}$ if there exists a pointed model $(\mathfrak{M}, w)$ based on a tree of clusters with root w such that

(a) $(\mathfrak{M}, w)$ realises $\mathtt{t}$,

(b) for every strict immediate successor $C(y)$ of $C(w)$ in $\mathfrak{M}$, there is some $x \in C(y)$ such that the triple realised by $(\mathfrak{M}, x)$ belongs to $\mathcal{R}$,

(c) every triple from $\mathcal{R}$ is realised by $(\mathfrak{M}, y)$, for some strict immediate successor $C(y)$ of $C(w)$.

The algorithm is shown in Fig. 2; it uses the procedure $\mathsf{realise}$ described in Fig. 3.

We show first that this 'algorithm' can be refined in such a way that it indeed runs in ExpSpace. After that we will show its completeness and correctness.

Lemma 12 *It can be decided in* ExpSpace *whether a triple (t, Γ, Δ) is realisable in a finite pointed model $(\mathfrak{M}, w)$ based on a cluster.*

Proof. It is enough to give a nondeterministic decision procedure which requires exponential space only. Clearly, if (t, Γ, Δ) is realisable in a model based on a cluster, then one can find such a model of size bounded by $2^{|\varphi_1|}$. Now, the algorithm guesses such a pointed model $(\mathfrak{M}, w)$ based on $(W, W \times W)$. To check whether it realises (t, Γ, Δ) construct the characteristic formula $\chi_{var(\varphi_1)}(\mathfrak{M}, w)$ by taking

$$\chi_{var(\varphi_1)}(\mathfrak{M}, w) \;=\; t^1_{\mathfrak{M}}(w) \wedge \Box\Big(\bigwedge_{x,y \in W} \big(t^1_{\mathfrak{M}}(x) \rightarrow \Diamond t^1_{\mathfrak{M}}(y)\big) \wedge \bigvee_{x \in W} t^1_{\mathfrak{M}}(x)\Big),$$

Input: a triple $\mathsf{t} = (t, \Gamma, \Delta)$, where t is a φ_1-type and Γ, Δ are sets of $\varphi_1 \wedge \varphi_2$-types. $\mathsf{realise}(\mathsf{t})$ returns 'true' iff t is realisable in a pointed model based on a cluster or there exists a set $\mathcal{R}$ of triples with $|\mathcal{R}| \leq 2^{|\varphi_1 \wedge \varphi_2|}$ such that

(1) for all $\mathsf{t}' \in \mathcal{R}$, $\Phi_{\mathsf{t}'} \subsetneq \Phi_{\mathsf{t}}$,

(2) for all $\mathsf{t}' \in \mathcal{R}$, $\mathsf{realise}(\mathsf{t}')$ returns 'true,'

(3) t is obtained in one step from $\mathcal{R}$.

The procedure $\mathsf{realise}_0$ is defined in the same way as $\mathsf{realise}$ with the exception that condition (1) is omitted. (In particular, it still calls $\mathsf{realise}$ in (2)).

Figure 3: The procedures $\mathsf{realise}(t, \Gamma, \Delta)$ and $\mathsf{realise}_0(t, \Gamma, \Delta)$.

where, as before, $t^1_{\mathfrak{M}}(x)$ is regarded as the conjunction of the formulas in the φ_1-type $t^1_{\mathfrak{M}}(x)$ realised by $(\mathfrak{M}, x)$. It is readily seen that (t, Γ, Δ) realises $(\mathfrak{M}, w)$ iff

- $t = t^1_{\mathfrak{M}}(w)$,
- Γ consists of all $\varphi_1 \wedge \varphi_2$-types s such that $s \wedge t \wedge \chi_{var(\varphi_1)}(\mathfrak{M}, w)$ is satisfiable,
- Δ consists of all $\varphi_1 \wedge \varphi_2$-types s such that $s \wedge t^1_{\mathfrak{M}}(y) \wedge \chi_{var(\varphi_1)}(\mathfrak{M}, y)$ is satisfiable, for some $y \in W$.

As the characteristic formulas above are of size not exceeding $|\varphi_1| 2^{3|\varphi_1|}$ and as satisfiability in **S4** is decidable in PSpace, this can be checked in ExpSpace. ❑

Lemma 13 (i) *A triple (t, Γ, Δ) with $\Gamma \neq \emptyset$ and $\Gamma \subseteq \Delta$ is obtained in one step from a set $\mathcal{R}$ of triples if there exists a set T of φ_1-types containing t such that the following holds:*

- *a $\varphi_1 \wedge \varphi_2$-type s is in Γ iff the formula*

$$\begin{aligned}
\delta(s, t, T, \mathcal{R}) \;=\; & q \wedge s \wedge t \wedge \Box(\neg q \to \Box \neg q) \wedge \\
& \Box(q \to \bigvee_{r \in T} r) \wedge \bigwedge_{r, r' \in T} \Box\big(r \wedge q \to \Diamond(r' \wedge q)\big) \wedge \\
& \Box\big(q \to \bigwedge_{(t', \Gamma', \Delta') \in \mathcal{R}} \Diamond(\neg q \wedge \bigvee_{s \in \Gamma'} s)\big) \wedge \\
& \Box(\neg q \to \bigvee_{(t', \Gamma', \Delta') \in \mathcal{R}} \bigvee_{s \in \Delta'} s)
\end{aligned}$$

  (with some fresh variable q) is satisfiable,

- *a $\varphi_1 \wedge \varphi_2$-type s belongs to Δ iff either it is in some Δ' with $(t', \Gamma', \Delta') \in \mathcal{R}$ or the formula $\delta(s, T, \mathcal{R})$ obtained from $\delta(s, t, T, \mathcal{R})$ by deleting the conjunct t is satisfiable.*

(ii) *It is decidable in* ExpSpace (*in the size of* φ_1, φ_2) *whether a triple is obtained in one step from a set* $\mathcal{R}$ *of triples of cardinality not exceeding* $2^{|\varphi_1 \wedge \varphi_2|}$.

Proof. (ii) follows immediately from (i) and the fact that **S4** satisfiability is decidable in PSpace. It remains to prove (i).

($\Rightarrow$) Suppose $\mathsf{t} = (t, \Gamma, \Delta)$ is obtained in one step from $\mathcal{R}$. Then there is a finite pointed model $(\mathfrak{M}, w)$ satisfying conditions (a)–(c) above. Let

$$T \;=\; \{t^1_{\mathfrak{M}}(x) \mid x \in C(w)\}.$$

We show that $s \in \Gamma$ iff the formula $\delta(s, t, T, \mathcal{R})$ is satisfiable. Suppose $s \in \Gamma$. Then there exists a finite pointed model $(\mathfrak{M}', w')$ with root w' which is $var(\varphi_1)$-bisimilar to $(\mathfrak{M}, w)$ and satisfies s at w'. Let $\mathfrak{M}' = (\mathfrak{F}', \mathfrak{V}')$, $\mathfrak{F}' = (W', R')$, and let $\sim$ be the $var(\varphi_1)$-bisimulation between $(\mathfrak{M}, w)$ and $(\mathfrak{M}', w')$. As q is a fresh variable, we may assume that

$$\mathfrak{V}'(q) \;=\; \{y \in W' \mid \exists z \in W' \exists x \in C(w)\; (yR'z \wedge z \sim x)\}.$$

Clearly, all of the conjuncts of $\delta(s, t, T, \mathcal{R})$, save, possibly,

$$\alpha \;=\; \Box\big(q \to \bigwedge_{(t',\Gamma',\Delta') \in \mathcal{R}} \Diamond(\neg q \wedge \bigvee_{s \in \Gamma'} s)\big)$$

are satisfied by $(\mathfrak{M}', w')$. Now, for every y such that $(\mathfrak{M}', y) \models q$ we have to ensure that $y \models \bigwedge_{(t',\Gamma',\Delta') \in \mathcal{R}} \Diamond(\neg q \wedge \bigvee_{s \in \Gamma'} s)$. Let $(t', \Gamma', \Delta') \in \mathcal{R}$. Take a z from $\mathfrak{M}'$ and a y' from $C(w)$ such that $yR'z$ and $z \sim y'$. Take a z' with $y'Rz'$ such that $(\mathfrak{M}, z')$ realises (t', Γ', Δ'). We find a z'' in $\mathfrak{M}'$ such that $z' \sim z''$ and $zR'z''$. Hence, $yR'z''$. Clearly, by the definition, there exists $s \in \Gamma'$ such that $z'' \models s$. Hence, $y \models \Diamond s$. But of course, we may have $z'' \models q$. To repair this 'defect' take a disjoint copy $\mathfrak{M}_{z''}$ of the submodel generated by z'' and make q false everywhere in it. Then we add the resulting model to $\mathfrak{M}'$ as a new successor of y. Clearly this can be done for all such defects, and the resulting model satisfies $\delta(s, t, T, \mathcal{R})$.

Conversely, suppose $(\mathfrak{N}, u) \models \delta(s, t, T, \mathcal{R})$. To prove that $s \in \Gamma$, we need a model $(\mathfrak{N}', u')$ which is $var(\varphi_1)$-bisimilar to $(\mathfrak{M}, w)$ and such that $(\mathfrak{N}', u') \models s$. We construct such a model by taking first all those points from $\mathfrak{N}$ where q holds, together with the induced quasi-order and valuation.

Next, for every point y in $\mathfrak{N}$ where q does not hold, we find a model $(\mathfrak{N}_y, y)$ as follows:

- If $(\mathfrak{N}, y) \models s'$ for some $s' \in \Gamma'$ and $(t', \Gamma', \Delta') \in \mathcal{R}$, then by condition (c) above, there is a strict immediate successor $C(v)$ of $C(w)$ in $\mathfrak{M}$ such that $(\mathfrak{M}, v)$ realises (t', Γ', Δ'). Then $(\mathfrak{N}_y, y)$ is a model which is $var(\varphi_1)$-bisimilar to $(\mathfrak{M}, v)$ and satisfies s' at y.

- If $(\mathfrak{N}, y) \not\models s'$ for any $s' \in \Gamma'$ with $(t', \Gamma', \Delta') \in \mathcal{R}$ then, by the last conjunct of $\delta(s, t, T, \mathcal{R})$, we have some $s' \in \Delta'$ with $(t', \Gamma', \Delta') \in \mathcal{R}$ such that $y \models s'$. Let v be a successor of w in $\mathfrak{M}$ and $(\mathfrak{N}_y, y)$ a model satisfying s' at y such that $(\mathfrak{M}, v)$ is $var(\varphi_1)$-bisimilar to $(\mathfrak{N}_y, y)$.

We may assume (by the third line in the definition of $\delta(s,t,T,\mathcal{R})$ and because we can add sufficiently many copies of a generated submodel of $(\mathfrak{N},u)$ to $(\mathfrak{N},u)$) that for every v in an immediate successor $C(v)$ of $C(w)$ in $\mathfrak{M}$ which realises some $(t',\Gamma',\Delta')\in\mathcal{R}$, there exists $s'\in\Gamma'$ such that there exists a y (in which q is false) such that $(\mathfrak{N}_y,y)$ is a model which is $var(\varphi_1)$-bisimilar to $(\mathfrak{M},v)$ and satisfies s' at y.

Now we add all such $\mathfrak{N}_y$ to the already selected points where q holds in such way that all points in an $\mathfrak{N}_y$-model are successors of an already chosen point x if, and only if, y was a successor of x. The resulting model $(\mathfrak{N}',u)$ is clearly $var(\varphi_1)$-bisimilar to $(\mathfrak{M},w)$ and satisfies s at u.

The corresponding claim for Δ is proved analogously.

($\Leftarrow$) Conversely, suppose that we are given $\mathcal{R}$, t and T satisfying the conditions of our lemma. We can also assume that, for every triple $\mathsf{t}'\in\mathcal{R}$, there is a model $(\mathfrak{M}_{\mathsf{t}'},w_{\mathsf{t}'})$ realising t'. Our aim is to show that the triple $\mathsf{t}=(t,\Gamma,\Delta)$ defined in the formulation of the lemma is obtained from $\mathcal{R}$ in one step.

We construct a model $(\mathfrak{M},w)$ satisfying (a)–(c) for t and $\mathcal{R}$ as follows. Since the formula $\delta(s,t,T,\mathcal{R})$ is satisfiable for $s\in\Gamma$, all of the φ_1-types in T contain the same box and diamond formulas from $\mathsf{sub}(\varphi_1)$. For each $t'\in T$, we take a point, say $x_{t'}$, and define a valuation in it by $x_{t'}\models p$ iff $p\in t'$. These points with the defined valuations will form the root cluster of $\mathfrak{M}$ with $w=x_t$. As immediate successors of $C(w)$ we add the (disjoint) models $(\mathfrak{M}_{\mathsf{t}'},w_{\mathsf{t}'})$ realising all $\mathsf{t}'\in\mathcal{R}$. Clearly the resulting model $(\mathfrak{M},w)$ satisfies conditions (b) and (c). Let us check that it satisfies (a), i.e., that $(\mathfrak{M},w)$ realises t.

The fact that, for every $\varphi\in\mathsf{sub}(\varphi_1)$ and every $t'\in T$, we have $\varphi\in t'$ iff $(\mathfrak{M},x_{t'})\models\varphi$ is proved by induction on the construction of ψ. Let us only consider the nontrivial case $\varphi=\Diamond\psi$. Let $\Diamond\psi\in t'$. If $\psi\in t''$ for some $t''\in T$, then we have $(\mathfrak{M},x_{t'})\models\Diamond\psi$ by IH. So assume that there is no such t''. As $\delta(s,t,T,\mathcal{R})$ is satisfied in some model $(\mathfrak{N},u)$, there is a point v in $\mathfrak{N}$ such that $v\models\neg q$ and $v\models\psi$. Let s' be the $\varphi_1\wedge\varphi_2$-type realised by $(\mathfrak{N},v)$. By the last conjunct of $\delta(s,t,T,\mathcal{R})$, there is $\mathsf{t}'=(t',\Gamma',\Delta')\in\mathcal{R}$ with $s'\in\Delta'$. But then all formulas $\Diamond\chi\in s'\cap\mathsf{sub}(\varphi_1)$ must be true at $w_{\mathsf{t}'}$ in $\mathfrak{M}_{\mathsf{t}'}$. Therefore, $(\mathfrak{M},x_{t'})\models\Diamond\psi$. Conversely, let $(\mathfrak{M},x_{t'})\models\Diamond\psi$ for some $x_{t'}$ from the root cluster. Suppose again that ψ is not true anywhere in this cluster (for otherwise $\Diamond\psi\in t'$ follows from IH). But then $\Diamond\psi\in t''\subseteq s'\in\Gamma''$ for some (t'',Γ'',Δ'') in $\mathcal{R}$ and all $s'\in\Gamma''$. As $\delta(s,t,T,\mathcal{R})$ is satisfiable, it follows that $\Diamond\psi\in t'$ for all $t'\in T$.

It follows that $(\mathfrak{M},w)$ satisfies t and that T is the set of φ_1-types satisfied in $C(w)$. But then we have shown in the proof of ($\Rightarrow$) above, that $\mathsf{t}=\mathsf{t}'$, where t' is the triple realised by $(\mathfrak{M},w)$. ❏

We can now analyse the algorithm in Fig. 2. By Lemmas 12 and 13 and condition (1) of the procedure $\mathsf{realise}$, the procedures $\mathsf{realise}$ and $\mathsf{realise}_0$ always terminate and require exponential space only.

Now suppose that $\varphi_1\wedge\varphi_2$ is not a conservative extension of φ_1. Then there is a realisable triple (t,Γ,Δ) such that $\varphi_1\in t$ but $\varphi_1\wedge\varphi_2\notin s$, for any $s\in\Gamma$. Take a pointed model $(\mathfrak{M},w)$ with the properties of Lemma 11 which realises a $\mathsf{t}'=(t,\Gamma',\Delta')$ with $\Gamma'\subseteq\Gamma$ and $\Delta'\subseteq\Delta$. Now, let the algorithm in Fig. 2 guess the triple t'. Then it obviously returns '$\varphi_1\wedge\varphi_2$ is a conservative extension of φ_1.' Observe that we start with the procedure $\mathsf{realise}_0$ instead of $\mathsf{realise}$ because we have not proved that $\Phi_{\mathsf{t}'} \supsetneq \Phi_{\mathsf{t}''}$ for every triple t'' realised is a strict successor of w.

In conclusion, we obtain:

Theorem 14 *The conservativeness problem for* **S4** *is decidable in* ExpSpace.

7 The upper bound for GL.3

The set of finite rooted frames for **GL.3** coincides with the set of finite strict linear orders (W, R). Observe that for models based on strict linear orders, every $\mathbf{p}$-bisimulation between $(\mathfrak{M}, w)$ and $(\mathfrak{M}', w')$ is an isomorphism between the submodels of these models generated by w and w', respectively, and restricted to the variables from $\mathbf{p}$.

A pair (t, Γ), where t is a φ_1-type t and Γ is a set of $\varphi_1 \wedge \varphi_2$-types, is said to be *realised in a pointed model* $(\mathfrak{M}, w)$ if

- $(\mathfrak{M}, w) \models t$ and
- Γ is the set of $\varphi_1 \wedge \varphi_2$-types s such that $s \wedge \chi_{var(\varphi_1)}(\mathfrak{M}, w)$ is satisfiable (in a finite model for **GL.3**).

Lemma 15 *$\varphi_1 \wedge \varphi_2$ is not a conservative extension of φ_1 in* **GL.3** *iff there exists a pointed model $(\mathfrak{M}, w)$ based on a strict linear order such that*

- *for the pair (t, Γ) realised by $(\mathfrak{M}, w)$, $\varphi_1 \in t$ and $\varphi_1 \wedge \varphi_2 \notin s$, for any $s \in \Gamma$,*
- *for any two points $x \neq y$ in $\mathfrak{M}$, the pair realised by $(\mathfrak{M}, x)$ is different from the pair realised by $(\mathfrak{M}, y)$.*

In particular, the length of the strict linear order underlying $\mathfrak{M}$ does not exceed $2^{2^{|\varphi_1 \wedge \varphi_2|}}$.

Proof. The proof is easy and left to the reader. ❑

We say that a pair of types (t, t') is *suitable* if

- $\Box\psi \in t$ implies $\psi, \Box\psi \in t'$, and
- $\neg\Box\psi \in t$ implies $\neg\psi \in t'$ or $\neg\Box\psi \in t'$.

The non-deterministic algorithm deciding non-conservativeness in **GL.3** is shown in Fig. 4. Clearly, this algorithms requires exponential space only. Using the fact that NExpSpace = ExpSpace, we obtain an ExpSpace algorithm. The correctness of this algorithm follows from Lemma 15.

Theorem 16 *The conservativeness problem for* **GL.3** *is decidable in* ExpSpace.

Input: formulas φ_1, φ_2.

1. Choose, non-deterministically, a pair (t, Γ), where t is a φ_1-type containing φ_1 and Γ is a set of $\varphi_1 \wedge \varphi_2$-types such that $t \subseteq s$ for all $s \in \Gamma$ and $\varphi_1 \wedge \varphi_2 \notin s$ for any $s \in \Gamma$.

2. Call $\mathsf{real}(t, \Gamma, 2^{2^{|\varphi_1 \wedge \varphi_2|}})$.

Definition of procedure $\mathsf{real}(t, \Gamma, n)$ (where n is a natural number coded in binary):

1. If no s in Γ contains a formula of the form $\neg\Box\psi$, then return '$\varphi_1 \wedge \varphi_2$ is not a conservative extension of φ_1' and stop. Else,

2. if $n = 0$, then return '$\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1' and stop. Else,

3. choose, non-deterministically, a pair (t', Γ'), where t' is a φ_1-type and Γ' is a set of $\varphi_1 \wedge \varphi_2$-types. Check whether (t', Γ') has the following properties:

   - $t' \subseteq s'$, for all $s' \in \Gamma$,
   - (t, t') is a suitable pair,
   - for every $s \in \Gamma$, there exists $s' \in \Gamma'$ such that (s, s') is a suitable pair,
   - for each $\varphi_1 \wedge \varphi_2$-type $s \notin \Gamma$, $t \not\subseteq s$ or there is no $s' \in \Gamma'$ such that (s, s') is a suitable pair.

4. If (t', Γ') does not have these properties, return '$\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1' and stop. Else,

5. set $n := n - 1$.

6. Call $\mathsf{real}(t', \Gamma', n)$.

Figure 4: Deciding non-conservativeness for **GL.3**.

8 The lower bound for GL.3

In this section we show that the conservativeness problem for **GL.3** is ExpSpace-hard. Let $\mathcal{M} = (Q, \Sigma, \Gamma, q_0, \Delta)$ be a Turing machine that solves an ExpSpace-hard problem and consumes at most 2^n tape cells if started on an input of length n.

Let $\mathsf{w} = a_0 \cdots a_{n-1} \in \Sigma^*$ be an input to $\mathcal{M}$. Our aim is to construct formulas φ_1 and φ_2 (depending on $\mathcal{M}$ and w) such that $\varphi_1 \wedge \varphi_2$ is *not* a conservative extension of φ_1 if, and only if, $\mathcal{M}$ does accept w. More precisely, we construct φ_1 and φ_2 in such a way that, if ψ is a witness for (φ_1, φ_2), then rooted models of $\varphi_1 \wedge \psi$ describe an accepting computation of $\mathcal{M}$ on w. In these models, each point represents a tape cell of a configuration of $\mathcal{M}$, and moving to the immediate successor of a point means moving to the next tape cell in the same configuration, or, if we are already at the end of the configuration, moving to the first tape cell of a successor configuration. Such models will have depth $m := 2^n \cdot 2^{2^n}$ since the length of computations is bounded by 2^{2^n}, each configuration has length $2^n - 1$, and any two consecutive configurations are connected by an additional edge.

We proceed in two steps. First, we construct formulas φ_1' and φ_2' such that, if ψ is a witness formula for (φ_1', φ_2') and we have $(\mathfrak{M}, w) \models \varphi_1' \wedge \psi$ (with $\mathfrak{M}$ based on a finite strict linear order (W, R)), then the following holds:

(P1) if $w_0, \dots, w_k \in W$ are such that $w_0 = w$ and w_i is the immediate predecessor of w_{i+1} for $i < k$, then

(a) the binary counter C realised via the propositional variables $c_0, \dots, c_{n-1}$ has the value $k \mod 2^n$ at w_k;

(b) the truth values of the propositional variable a at the worlds $w_0, \dots, w_k$ describe the first $k+1$ bits of the consecutive values of a 2^n-bit counter A starting at 0 and counting modulo $2^{2^n} - 1$;

(c) if $w_i, \dots, w_{i+\ell}$ is a subsequence of $w_0, \dots, w_k$, $\ell < 2^n$, such that C has value 0 at w_i, then the propositional variable z is true at $w_{i+\ell}$ iff a is false in at least one of $w_0, \dots, w_{i+\ell-1}$;

(d) if $k < m$, then there is a w' with $w_k R w'$.

Intuitively, φ_1' and φ_2' generate the structure into which we will accomodate computations of the Turing machines, but they do not describe these computations themselves. In the second step, we will add further conjuncts to φ_1' and φ_2' to obtain φ_1 and φ_2 that encode computations of $\mathcal{M}$ on w as described above.

The formula φ_1' is simply defined as

$$\varphi_1' \quad = \quad (C = 0) \wedge (a \vee \neg a)$$

where $(C = 0)$ is the obvious formula stating that the binary counter composed from the bits $c_0, \dots, c_{n-1}$ has value 0, while the second conjunct only serves the purpose of introducing a into the signature of φ_1'.

The formula φ_2' is the conjunction of all formulas in Fig. 5. Intuitively, φ_2' says that models of $\varphi_1' \wedge \varphi_2'$ do not satisfy (P1). In conjunct (5), we decide whether property (a), (b), (c) or (d) of (P1) is violated. If (a) is violated, then we mark the place where this happens with the variable d in (6): when going from the immediate predecessor of d to d, the counter C is not properly incremented. There are two ways in which incrementation can fail: first, there may be no 0-bit lower than the i-bit, but the i-th bit is not toggled. Second, there may be a 0-bit lower than the i-th bit, but the i-th bit is toggled. These two cases are distinguished via d_1 and d_2 in (6), and then implemented in (7) and (8).

Violation of (b) is treated in (9) and rests on the behaviour of the propositional letters m, m', and m'' that is axiomatised by (16)–(23). The point that witnesses the diamond in (9) marks the place where the counter A fails to increment properly. The idea is that we set m to true, store the value of the counter C in $s_0, \dots, s_{n-1}$, and ensure that the latter variables have the same value everywhere in the future. Similarly, the value of a (the truth value of A) is stored in s and propagated into the future. Then, the axiomatisation of m, m', m'' ensures that m holds until the next ($C = 0$) point, where it changes to m'. Then m' holds until the next ($C = 0$) point. After that m'' holds forever. Thus, we can identify the identical bit in the next counter value by looking for the place where m' is true and $c_0, \ldots, c_{n-1}$ agree with $s_0, \ldots, s_{n-1}$. It remains to distinguish the two cases for incrementation failure as in the case of C. This is done with the help of the variable z. Due to (c), we can assume that z tells us at the witness point whether or not we have seen a zero in the counter A before. Note that we can assume that (c) is satisfied since, if it is not, then (P1) is violated anyway.

Violation of (c) is described in (10)–(14) in a straightforward way (distinguishing four different kinds of violations), and violation of (d) is described in (15), again assuming that property (c) is satisfied.

Now for the extension of φ_1' to φ_1 and of φ_2' to φ_2. We start with φ_1 whose additional conjuncts are shown in Fig. 6. The purpose of (24)–(29) is simply to enforce some basic things about Turing machines: there is exactly one symbol per tape cell (24), there are never two different states per tape cell (26), and there is at most one cell labelled with a state in each configuration (27). Note that (27) is based on the assumption that in each configuration, h is true on all cells of a configuration that are are (directly or indirectly) following the cell where the head is. Formulas (28)–(29) state that the computation starts in the initial state q_0 and that the tape is initially labelled with w followed by blanks. Here, it is assumed that f is true throughout the first configuration. The assumptions concerning h and f will be established via φ_2. We assume that our Turing machines always end up in the accepting state q_a or rejecting state q_r. Thus, (30) ensures that the computation is accepting.

The purpose of the additional conjuncts of φ_2 is to ensure that the variables h and f behave as expected, that the Turing machine moves according to the transition relation, and that tape cells that are not under the head do not change when the machine makes a step. More precisely, we define φ_2 in such a way that, if ψ is a witness formula for (φ_1, φ_2) and we have $(\mathfrak{M}, w) \models \varphi_1 \wedge \psi$, then (P1) above together with the following properties (P2)–(P4) are satisfied:

(P2) If $w_0, \ldots, w_k \in W$ are such that $w_0 = w$ and w_i is the immediate predecessor of w_{i+1} for $i < k$, then

  (e) if $w_i, \ldots, w_{i+\ell}$ is a subsequence of $w_0, \ldots, w_k$, $\ell < 2^n$, such that C has value 0 at w_i, $i \leq j \leq i + \ell$, and $(\mathfrak{M}, w_j) \models m_q$ for some $q \in Q$, then $(\mathfrak{M}, w_d) \models h$ for all d with $j < d < i + \ell$;

  (f) if $k < 2^n$, then $(\mathfrak{M}, w) \models f$.

(P3) The Turing machine moves according to its transition table: if w' is accessible from w such that $(\mathfrak{M}, w') \models m_q \wedge m_a$ and $v_0, \ldots, v_{2^n+1}$ is the outgoing path in $\mathfrak{M}$ starting from w', then one of the following holds:

- there is $(q', b, L) \in \delta(q, a)$ such that $(\mathfrak{M}, v_{2^n}) \models m_b$ and $(\mathfrak{M}, v_{2^n-1}) \models m_{q'}$;
- there is $(q', b, R) \in \delta(q, a)$ such that $(\mathfrak{M}, v_{2^n}) \models m_b$ and $(\mathfrak{M}, v_{2^n+1}) \models m_{q'}$.

$$p_a \vee p_b \vee p_c \vee p_d \tag{5}$$

$$p_a \to \Diamond^+(\Diamond d \wedge \neg\Diamond\Diamond d \wedge (d_1 \vee d_2)) \tag{6}$$

$$p_a \to \Box^+\Big(d_1 \to \bigvee_{i<2^n} \Big(((c_i \wedge \Box(d \to c_i)) \vee (\neg c_i \wedge \Box(d \to \neg c_i))) \wedge \bigwedge_{j<i} c_j\Big)\Big) \tag{7}$$

$$p_a \to \Box^+\Big(d_2 \to \bigvee_{i<2^n} \Big(((c_i \wedge \Box(d \to \neg c_i)) \vee (\neg c_i \wedge \Box(d \to c_i))) \wedge \bigvee_{j<i} \neg c_j\Big)\Big) \tag{8}$$

$$\begin{aligned} p_b \to \Diamond^+\Big(m \wedge \bigwedge_{i<2^n} ((c_i \to \Box s_i) \wedge (\neg c_i \to \Box\neg s_i)) \\ \wedge (a \to \Box s) \wedge (\neg a \to \Box\neg s) \\ \wedge \Big((\neg z \wedge \Diamond(m' \wedge (a \leftrightarrow s) \wedge \bigwedge_{i<2^n} (c_i \leftrightarrow s_i))) \\ \vee (z \wedge \Diamond(m' \wedge (a \leftrightarrow \neg s) \wedge \bigwedge_{i<2^n} (c_i \leftrightarrow s_i)))\Big)\Big) \end{aligned} \tag{9}$$

$$p_c \to p_{c_1} \vee p_{c_2} \vee p_{c_3} \vee p_{c_4} \tag{10}$$

$$p_{c_1} \to \Diamond^+((C = 0) \wedge z) \tag{11}$$

$$p_{c_2} \to \Diamond^+(\Diamond d \wedge \neg\Diamond\Diamond d \wedge (d_1 \vee d_2) \wedge \neg a \wedge (C < 2^n - 1) \wedge \Box(d \to \neg z)) \tag{12}$$

$$p_{c_3} \to \Diamond^+(\Diamond d \wedge \neg\Diamond\Diamond d \wedge (d_1 \vee d_2) \wedge z \wedge (C < 2^n - 1) \wedge \Box(d \to \neg z)) \tag{13}$$

$$p_{c_4} \to \Diamond^+(\Diamond d \wedge \neg\Diamond\Diamond d \wedge (d_1 \vee d_2) \wedge \neg z \wedge a \wedge (C < 2^n - 1) \wedge \Box(d \to z)) \tag{14}$$

$$p_d \to \Diamond^+(\Box\bot \wedge ((C < 2^n - 1) \vee z \vee \neg a)) \tag{15}$$

$$\Box^+(\neg(m \wedge m') \wedge \neg(m \wedge m'') \wedge \neg(m' \wedge m'')) \tag{16}$$

$$\Box^+(m \to \Box(m \vee m' \vee m'')) \tag{17}$$

$$\Box^+(m' \to \Box(m' \vee m'')) \tag{18}$$

$$\Box^+(m'' \to \Box m'') \tag{19}$$

$$\Box^+\big((m \wedge (C = 2^n - 1)) \to \Box(m' \vee m'')\big) \tag{20}$$

$$\Box^+\big((m' \wedge (C = 2^n - 1)) \to \Box m''\big) \tag{21}$$

$$\Box^+\big((m \wedge \neg\Diamond m) \to (C = 2^n - 1)\big) \tag{22}$$

$$\Box^+\big((m' \wedge \neg\Diamond m') \to (C = 2^n - 1)\big) \tag{23}$$

Figure 5: The conjuncts of φ_2'.

$$\Box^+\Big(\bigvee_{a\in\Sigma} m_a \wedge \bigwedge_{a,b\in\Sigma \text{ with } a\neq b} \neg(m_a \wedge m_b)\Big) \tag{24}$$

$$\Box^+ \bigwedge_{q,q'\in Q \text{ with } q\neq q'} \neg(m_q \wedge m'_q) \tag{25}$$

$$\Box^+\neg(h \wedge \bigvee_{q\in Q} m_q) \tag{26}$$

$$f \wedge m_{q_0} \wedge m_{a_0} \tag{27}$$

$$\Box^+(((C = i) \wedge f) \to m_{a_i}) \text{ for } 1 \leq i < n \tag{28}$$

$$\Box^+(((C \geq n) \wedge f) \to m_{\mathsf{blank}}) \tag{29}$$

$$\Diamond^+ m_{q_r} \tag{30}$$

Figure 6: Additional conjuncts for φ_1.

(P4) The contents of those cells that are not under the head does not change in the next configuration: if w' is accessible from w, $(\mathfrak{M}, w') \not\models \bigvee_{q\in Q} m_q$, and $v_0, \ldots, v_{2^n}$ is the outgoing path in $\mathfrak{M}$ starting from w', then $(\mathfrak{M}, w') \models m_a$ implies $(\mathfrak{M}, v_{2^n}) \models m_a$, for all $a \in \Sigma$.

More precisely, we obtain φ_2 from φ'_2 by replacing (5) in Fig. 5 with the disjunction

$$p_a \vee p_b \vee p_c \vee p_d \vee p_e \vee p_f \vee p_3 \vee p_4 \tag{$*$}$$

and then adding all formulas in Fig. 7 as conjuncts.

Recall that a model of φ_1 should be extendable to a model of φ_2 if it does *not* satisfy (P1)–(P3). Similar to (5) of φ'_2, $(*)$ selects the property to be violated. Violation of part (e) of (P2) is enforced by (31) to and (33) in a straightforward way. Even easier, (34) enforces part (f) of (P2). In (35)–(39), we enforce (P3). The idea is similar to that of enforcing part (b) of (P1) and, in particular, also relies on the proper behaviour of the variables m, m', and m''. Finally, (39) enforces violation of (P4). The idea is again similar.

We thus obtain the following:

Lemma 17 *$\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1 in* **GL.3** *iff $\mathcal{M}$ accepts* w.

By the choice of the Turing machine $\mathcal{M}$, it follows that we have

Theorem 18 *The conservativeness problem for* **GL.3** *is* ExpSpace*-hard.*

$$
\begin{aligned}
p_e &\to p_{e_1} \vee p_{e_2} && (31)\\
p_{e_1} &\to \Diamond^+\big(\bigvee_{q\in Q} m_q \wedge \Diamond(d \wedge \neg h \wedge \neg(C=0)) \wedge \neg\Diamond\Diamond d\big) && (32)\\
p_{e_2} &\to \Diamond^+\big(h \wedge \Diamond(d \wedge \neg h \wedge \neg(C=0)) \wedge \neg\Diamond\Diamond d\big) && (33)\\
p_f &\to \Diamond^+\big(f \wedge \Diamond(d \wedge \neg f \wedge \neg(C=0)) \wedge \neg\Diamond\Diamond d\big) && (34)\\
p_3 &\to \Diamond^+\Big(m \wedge \bigwedge_{i<2^n} \big((c_i \to \Box s_i) \wedge (\neg c_i \to \Box\neg s_i)\big) && (35)\\
&\qquad \wedge \bigvee_{a\in\Sigma} \big(m_a \wedge \bigvee_{q\in Q} (m_q \wedge \bigwedge_{(q',b,M)\in\delta(q,a)} x_{(q',b,M)})\big)\Big)\\
p_3 &\to \Box^+(x_{(q',b,M)} \to \Diamond\big(m' \wedge \bigwedge_{i<2^n} (c_i \leftrightarrow s_i) \wedge (\neg m_b \vee y_{(q',b,M)}\big) && (36)\\
p_3 &\to \Box^+(y_{(q',b,R)} \to (\Diamond y_b \wedge \neg\Diamond\Diamond y_b)) && (37)\\
p_3 &\to \Box^+(y_b \to m_b) && (38)\\
p_3 &\to \Box^+((\Diamond y_{(q',b,L)} \wedge \neg\Diamond\Diamond y_{(q',b,L)}) \to m_b && (39)\\
p_4 &\to \Diamond^+\big(\bigwedge_{q\in Q} \neg m_q \wedge m \wedge \bigwedge_{i<2^n} ((c_i \to \Box s_i) \wedge (\neg c_i \to \Box\neg s_i))\big) && (40)\\
&\qquad \wedge \bigwedge_{a\in\Sigma} (m_a \to \Box s_a)\\
&\qquad \wedge \Diamond(m' \wedge \bigwedge_{i<2^n} (c_i \leftrightarrow s_i) \wedge \bigwedge_{a\in\Sigma} (s_a \to \neg m_a))
\end{aligned}
$$

Figure 7: Additional conjuncts for φ_2.

9 Discussion

We have investigated the complexity of the conservativeness problem for the local consequence relation of a number of basic modal logics. One interesting conclusion is that the complexity of deciding conservativeness is not monotonically related to the complexity of the logic in question: for example, the satisfiability problem is NP-complete for **GL.3** and PSpace-complete for **K**, while the conservativeness problem is (probably) more complex for **GL.3** than for **K**. This resembles the situation with products of modal logics where **GL.3** $\times$ **GL.3** is Π^1_1-complete [11], while **K** $\times$ **K** is decidable [3].

In this paper, we have considered modal languages with one modal operator only. It is not difficult, however, to modify the proofs above to show that conservativeness (for the local consequence relation) is still NExpTime-complete for multimodal **S5** and multimodal **K**. Similarly, for multimodal **S4** and **K4** it is still decidable in ExpSpace.

For the global consequence relation the results are different: recall that φ *follows globally* from ψ in a modal logic L if φ is true everywhere in a model based on a frame for L whenever ψ is true everywhere in this model. Conservativeness with respect to the global consequence relation is now defined in the obvious way. Of course, for m-transitive modal logics the complexity upper bound for deciding conservativeness with respect to the local consequence is an upper bound for deciding conservativeness relative to the global consequence relation as well. This applies to **S5**, **S4** and **GL.3**. For **K**, however, deciding conservativeness with respect to the global consequence becomes 2ExpTime-complete, as follows from the investigation of conservative extensions in description logics in [5]. We expect deciding conservativeness with respect to the global consequence in multimodal **S5** and **S4** to be 2ExpTime-complete as well.

References

[1] G. Antoniou and K. Kehagias. A note on the refinement of ontologies. *International Journal of Intelligent Systems*, 15:623–632, 2000.

[2] A. Chagrov and M. Zakharyaschev. *Modal Logic*. Oxford University Press, Oxford, 1997.

[3] D. Gabbay and V. Shehtman. Products of modal logics. Part I. *Logic Journal of the IGPL*, 6:73–146, 1998.

[4] S. Ghilardi. An algebraic theory of normal forms. *Annals of Pure and Applied Logic*, 71(3):189–245, 1995.

[5] S. Ghilardi, C. Lutz, and F. Wolter. Did I damage my ontology? A case for conservative extensions in description logics. In *Proceedings of the International Conference of Principles of Knowledge Representation and Reasoning*, 2006.

[6] S. Ghilardi and M. Zawadowski. Undefinability of propositional quantifiers in the modal system S4. *Studia Logica*, 55(2):259–271, 1995.

[7] V. Goranko and M. Otto. Modal model theory. In J. van Benthem, P. Blackburn, and F. Wolter, editors, *Handbook of Modal Logic*. Elsevier, 2006.

[8] M. Kracht. Modal consequence relations. In J. van Benthem, P. Blackburn, and F. Wolter, editors, *Handbook of Modal Logic*. Elsevier, 2006.

[9] C. Papadimitriou. *Computational Complexity*. Addison Wesley, 1995.

[10] A. Pitts. On an interpretation of second-order quantification in first-order intuitionistic propositional logic. *Journal of Symbolic Logic*, 57(1):33–52, 1992.

[11] M. Reynolds and M. Zakharyaschev. On the products of linear modal logics. *Journal of Logic and Computation*, 11:909–931, 2001.

[12] B. ten Cate, W. Conradie, M. Marx, and Y. Venema. Definitorially complete description logics. In *Proceedings of the International Conference of Principles of Knowledge Representation and Reasoning*, 2006.

[13] W.M. Turski and T. Maibaum. *The Specification of Computer Programs.* Addison-Wesley, 1987.

[14] P. van Emde Boas. The convenience of tilings. In A. Sorbi, editor, *Complexity, Logic and Recursion Theory*, volume 187 of *Lecture Notes in Pure and Applied Mathematics*, pages 331–363. Marcel Dekker Inc., 1997.

[15] A. Visser. Uniform interpolation and layered bisimulation. In *Gödel'96 (Brno, 1996)*, volume 6 of *Lecture Notes Logic*, pages 139–164. Springer, Berlin, 1996.

[16] F. Wolter and M. Zakharyaschev. Modal decision problems. In J. van Benthem, P. Blackburn, and F. Wolter, editors, *Handbook of Modal Logic*. Elsevier, 2006.