4

How Many Variables Does One Need to Prove PSPACE-hardness of Modal Logics?

A. V. Chagrov and M. N. Rybakov

ABSTRACT. The main result of the paper is that the decision problem for the variable-free fragments of both **K** and **K4** is PSPACE-complete. It is also proved that the decision problem for one-variable fragments of **S4**, **Grz**, and **GL** is PSPACE-complete too. The questions about complexity of variable-free, one-variable, and two-variable fragments of some closed logics are discussed.

1 Introduction

It is well known that the decision problem for propositional modal logics **K**, **T**, **S4**, and many others is PSPACE-complete [6] (a good reference on complexity theory is [9]). Ladner's [6] proof of PSPACE-completeness of the decision problem for these logics was essentially based upon the fact that the modal language contains infinitely many propositional variables; he did not try to use as few of them as possible. On the other hand, the fragments with only finitely many variables are rather natural, and in many cases such a restriction may result in a lower algorithmic complexity: for example, classical propositional (non-modal) logic with finitely many variables is decidable in polynomial time.

About 15 years ago, an analysis of Ladner's [6] and other similar proofs gave the first author some (shaky) grounds to conjecture[2] that an infinite set of variables is really needed to prove PSPACE-completeness of modal logics, that finite variable fragments can really be of lower complexity. However, in the 1990s this conjecture was refuted: Halpern [5] showed that a sole variable is enough to prove PSPACE-hardness of the satisfiability problem for modal logics considered in [6]. And even before that Spaan [8] proved the 'Single Variable Reduction Theorem' (Theorem 5.4.6 of [8]) claiming that the satisfiability problem for modal formulas is polynomially reducible to the satisfiability problem for one-variable modal formulas.

[2]This conjecture was never published, but it was discussed at several conferences.

Advances in Modal Logic, Volume 4, 71–82.

In this connection, it is natural to raise the question on the complexity of the *variable-free* fragments of standard modal logics: is it possible to obtain optimal lower complexity bounds (PSPACE-hardness or NP-hardness) for the decision problem for modal logics using only *variable-free formulas*? Some results of this sort were proved by Spaan [8] (see Theorem 5.4.7 of [8]) — but for logics with two modal operators. On the other hand, note that Halpern's [5] language contains no constants ($\bot$ and $\top$), and so the obtained results for the one-variable fragment may be regarded as optimal (although the formulas from [5] that establish PSPACE-hardness of the satisfiability problem for **K** can be easily transformed into variable-free formulas with the same property by some substitution).

Our main aim in this paper is to find out the *minimal number* of variables one needs to prove PSPACE-hardness of the decision problem for standard modal logics. Since this problem (even in the full infinite language) is in PSPACE, we thereby shall get PSPACE-completeness of the corresponding finite-variable (or even variable-free) fragments. For some of the logics considered — **T**, **S4**, **Grz**, **GL** — these fragments contain a sole variable, while variable-free fragments of these logics are decidable by polynomial algorithms. On the other hand, **K** and **K4** do have very expressive PSPACE-hard variable-free fragments.

In conclusion, we consider the satisfiability problem for some other logics. We also obtain a corollary on the frame complexity function for the considered fragments.

Our notation and notions are mostly standard: for modal logics and their complexity problems the reader can consult [3] or [10], for complexity theory see [4] and [9].

2 Main results

In this section we consider normal modal logics located between **K** and **K4**. For our aims it is enough to show PSPACE-hardness of the satisfiability (or, equivalently, the decision) problem for fragments we are interested in. First, we will use only variable-free formulas for proving PSPACE-hardness of the satisfiability problem for **K4**.

To begin with, we prove PSPACE-hardness of the satisfiability problem for **K4** in the *full language* that contains the propositional variables $p_1, p_2, p_3, \ldots$, the Boolean constant $\bot$ ('falsehood'), the Boolean connectives $\wedge$, $\vee$, $\rightarrow$, and the modal operator $\Box$. The constant $\top$ and the connectives $\neg$ and $\Diamond$ are used as standard abbreviations.

To prove PSPACE-hardness of the satisfiability problem for **K4**, we construct a reduction of the satisfiability problem for quantified Boolean formulas (which is known to be PSPACE-complete, see [9]) to the satisfiability problem for **K4**. Without loss of generality, we will confine ourselves to considering quantified Boolean formulas of the form $\varphi = Q_1 p_1 \dots Q_n p_n \varphi'$, where $Q_1, \dots, Q_n \in \{\forall, \exists\}$ and φ' is a quantifier-free formula with $p_1, \dots, p_n$ as its variables. Our reduction is a modification of the transformation considered in [6] (it can also be found in [2]). Given a formula $\varphi = Q_1 p_1 \dots Q_n p_n \varphi'$, we take additional variables $q_0, \dots, q_{n+1}$ and set

$$
\begin{aligned}
A &= \bigwedge_{i=1}^{n+1} (q_i \to q_{i-1}),\\
B &= \bigwedge_{i=1}^{n} [q_i \to (p_i \to \Box(q_i \to p_i)) \wedge (\neg p_i \to \Box(q_i \wedge \neg q_{n+1} \to \neg p_i))],\\
C &= \bigwedge_{\{i\,:\,Q_{i+1}=\forall\}} [q_i \wedge \neg q_{i+1} \to \Diamond(q_{i+1} \wedge \neg q_{i+2} \wedge p_{i+1}) \wedge \\
&\qquad\qquad \Diamond(q_{i+1} \wedge \neg q_{i+2} \wedge \neg p_{i+1})],\\
D &= \bigwedge_{\{i\,:\,Q_{i+1}=\exists\}} [q_i \wedge \neg q_{i+1} \to \Diamond(q_{i+1} \wedge \neg q_{i+2})].
\end{aligned}
$$

Let

$$\varphi^* = q_0 \wedge \neg q_1 \wedge \Box^+ A \wedge \Box^+ B \wedge \Box^+ C \wedge \Box^+ D \wedge \Box^+ (q_n \wedge \neg q_{n+1} \to \varphi'),$$

where $\Box^+\psi = \psi \wedge \Box\psi'$. Note that φ^* is constructed from φ by a polynomial time algorithm in the length of φ. Moreover,

(1) $\quad \varphi$ is true $\iff$ φ^* is **K4**-satisfiable,

and hence we obtain PSPACE-hardness of the satisfiability problem for **K4**; accordingly to Savitch's theorem [7] coPSPACE=PSPACE therefore we also obtain PSPACE-hardness of the decision problem for **K4**.

To prove ($\Rightarrow$) of (1), observe that the structure of the true formula φ clearly shows how to construct the required **K4**-model. As the underlying **K4**-frame we take the transitive reflexive tree of height $n+1$ such that if $Q_i = \forall$ then from every point of level $i-1$ (the root is of level 0) two points of level i are accessible and if $Q_i = \exists$ then one point of level i is accessible. The valuation is defined in the following way: q_i is true at all points of level greater than or equal to i, p_i is false at all points of level less than i; if a and b are two distinct points of level i accessible from some point of level $i-1$, then p_i is true at one of them, say a, and all points accessible from it, and p_i is false at the other — i.e., b — and all points accessible from it; if a is a point of level $i-1$ and b is the unique point of level i accessible from a, then if for the truth-values of $p_1, \dots, p_{i-1}$ at a for $\exists p_i Q_{i+1} p_{i+1} \dots Q_n p_n \varphi'$ to be true the 'true' was chosen for p_i, then we define p_i to be true at b

and at all points accessible from b, and if 'false' was chosen for p_i, then we make p_i false at b and at all points accessible from b as well. It should be clear that the formula φ' is true at every point of level n. Therefore, $\Box(q_n \wedge \neg q_{n+1} \to \varphi')$ is true at the root. It is easy to check that all other conjuncts of φ^* are true at the root. Hence φ^* is satisfied in the constructed **K4**-model.

($\Leftarrow$) If φ^* is true at a point a of some **K4**-model, then the conjuncts of φ^* allow us to extract in a 'step-by-step' manner a submodel of the given model in almost the same way as in the proof of ($\Rightarrow$), except that some points may be irreflexive. The formula φ' will be true at all points of level n in this model and the collections of the truth-values of $p_1, \ldots, p_n$ at points of level n provide us with a representative sample showing the truth of φ.

For any m, let

$$\alpha_m = \Box(\Diamond^m \Box\bot \wedge \neg\Diamond^{m+1}\Box\bot \to \Box(\Diamond\top \to \Diamond\Box\bot))$$

and denote by φ^*_α the formula obtained from φ^* by substituting $\alpha_1, \ldots, \alpha_{2n+2}$ for $p_1, \ldots, p_n, q_0, \ldots, q_{n+1}$, respectively.

LEMMA 1 *The formula φ^*_α is computed from φ^* by a polynomial time algorithm, and φ^*_α is* **K4***-satisfiable iff φ^* is* **K4***-satisfiable.*

Proof. Suppose φ^* is not **K4**-satisfiable. Then $\neg\varphi^* \in$ **K4**. But $\neg\varphi^*_\alpha$ is a substitution of $\neg\varphi^*$, from which it follows that $\neg\varphi^*_\alpha \in$ **K4**, and hence φ^*_α is not **K4**-satisfiable.

Let φ^* be **K4**-satisfiable. It means (see the proof of (1)) that the quantified Boolean formula φ is true. Hence φ^* is satisfied at the root w_0 of a reflexive and transitive tree $M = \langle W, R, v\rangle$ of height $n+1$. It is easy to see that the valuation v is hereditary: for every variable p and for any $w', w'' \in W$ such that $w'Rw''$ and $(M, w') \models p$, we have $(M, w'') \models p$.

Now we extend the model $M = \langle W, R, v\rangle$ to a model $M' = \langle W', R', v'\rangle$ in such a way that $(M', w_0) \models \varphi^*_\alpha$ will hold.

First observe that for α_m to be refuted at a point of a (reflexive and transitive) model, it is sufficient that the frame, depicted on the left-hand side of Fig. 1, be accessible from that point (black points of the frame are irreflexive, white ones are reflexive, and the accessibility relation is transitive). Denote this frame by F_m.

It is clear that if α_m is true at a point of a transitive model then it is true at all points accessible from it (the main connective of α_m is $\Box$). But if $k \neq m$ then $F_m \models \alpha_k$. Using these observations, we construct the required model M'. The construction takes $(2n+3)$ steps.

Step 0. Put $W_0 = W$, $R_0 = R$.

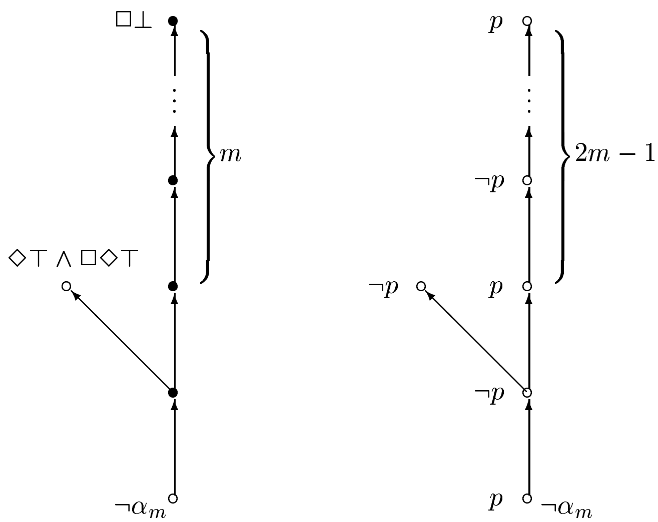

Figure 1. Frame F_m and model M_m.

Step m, where $1 \leq m \leq 2n+2$. Consider the set X_m of points in M where p_m is false. For every $w \in X_m$, we extend W_{m-1} and R_{m-1} so that a copy of F_m be accessible from w. Denote the resultant set of points by W_m and the transitive closure of the obtained accessibility relation by R_m.

Let $W' = W_{2n+2}$, $R' = R_{2n+2}$ and let v be an arbitrary valuation. For every ψ with $p_1, \ldots, p_n, q_0, \ldots, q_{n+1}$ we denote by ψ_α the formula obtained from ψ by substituting $\alpha_1, \ldots, \alpha_{2n+2}$ for $p_1, \ldots, p_n, q_0, \ldots, q_{n+1}$, respectively. Then for every subformula ψ of the (quantifier-free) Boolean formula φ' and every $w \in W$ of level n, the following equivalence holds:

$$(M', w) \models \psi_\alpha \iff (M, w) \models \psi.$$

This claim is proved by induction on the construction of ψ.

The most difficult case is the basis of induction. If $\psi = p_m$ then $\psi_\alpha = \alpha_m$ and we must prove that for every $w \in W$ of level n

$$(M', w) \models \alpha_m \iff (M, w) \models p_m.$$

Let w be a point of level n such that $(M, w) \not\models p_m$. Then at step m of the construction of M', model M was extended with a copy of F_m accessible from w. But then $(M', w) \not\models \alpha_m$.

Let $(M', w) \not\models \alpha_m$, where w is a point of M of level n. It is easy to see that a copy of F_m must be accessible from w. This is possible only if $(M, w) \not\models p_m$.

The induction step is trivial.

Thus, if w is a point in M of level n then $(M', w) \models \varphi'_\alpha$. Observe that the formula $\alpha_{2n+1} \wedge \neg\alpha_{2n+2}$ is true at some point w of M' only if w is a point of level n in M, because points in M validating $q_n \wedge \neg q_{n+1}$ are points of level n. Therefore, they are points of M' from which copies of F_{2n+2} are accessible, but copies of F_{2n+1} are not. So, at every other point of M' the formula $\alpha_{2n+1} \wedge \neg\alpha_{2n+2}$ is not true. Hence $(M', w_0) \models \Box(\alpha_{2n+1} \wedge \neg\alpha_{2n+2} \to \varphi'_\alpha)$.

It is easy to check that

$$(M', w_0) \models \alpha_{n+1} \wedge \neg\alpha_{n+2} \wedge \Box^+ A_\alpha \wedge \Box^+ B_\alpha \wedge \Box^+ C_\alpha \wedge \Box^+ D_\alpha.$$

It follows that $(M', w_0) \models \varphi^*_\alpha$. ■

Now we show how to modify φ^* and $\alpha_1, \ldots, \alpha_{2n+2}$ in order to prove similar results for some other logics. Consider a logic L such that $\mathbf{K} \subseteq L \subseteq \mathbf{K4}$. For any ψ, let

$$\Box^{\leq 1}\psi = \Box\psi; \quad \Box^{\leq k+1}\psi = \Box^{\leq k}\psi \wedge \Box^{\leq k+1}\psi.$$

Denote by $tr_k(\psi)$ the formula obtained from ψ by replacing each subformula $\Box\delta$ of ψ with $\Box^{\leq k}\delta$. Then one can readily prove that

$$\begin{aligned} \varphi \text{ is true} \quad &\iff \quad \varphi^* \text{ is } \mathbf{K4}\text{-satisfiable} \\ &\iff \quad tr_n(\varphi^*) \text{ is } L\text{-satisfiable.} \end{aligned}$$

Given any natural m, we take again the variable-free formula

$$\alpha_m = \Box(\Diamond^m\Box\bot \wedge \neg\Diamond^{m+1}\Box\bot \to \Box(\Diamond\top \to \Diamond\Box\bot)).$$

Denote by $tr_n(\varphi^*)_\alpha$ the formula obtained from $tr_n(\varphi^*)$ by substituting $\alpha_1, \ldots, \alpha_{2n+2}$ for $p_1, \ldots, p_n, q_0, \ldots, q_{n+1}$, respectively.

LEMMA 2 *The formula $tr_n(\varphi^*)_\alpha$ is constructed from φ^* by a polynomial time algorithm, and $tr_n(\varphi^*)_\alpha$ is L-satisfiable iff φ^* is* **K4***-satisfiable.*

Using the complexity properties of quantified Boolean formulas and their *-images, we obtain from Lemmas 1 and 2 the following:

THEOREM 3 *For every logic L from the interval $\mathbf{K} \subseteq L \subseteq \mathbf{K4}$, the satisfiability (and so the decision) problem for its variable-free fragment is* PSPACE*-hard.*

Ladner [6] gave polynomial space decision algorithms for **K**, **T**, and **S4**. A slight modification of Ladner's algorithm for **S4** provides a similar decision algorithm for **K4**. Hence we obtain

COROLLARY 4 *The satisfiability (and so the decision) problem for the variable-free fragments of* **K** *and* **K4** *are* PSPACE-*complete.*

Since the variable-free fragments of **S4**, **Grz**, **GL** are decidable in polynomial time (see Theorem 9 below), let us consider their one-variable fragments.

REMARK Halpern [5] proved that the decision problem for the one-variable fragment of **S4** is PSPACE-hard. But [5] contains some inaccuracies. Namely, the proof uses the following one-variable formulas:

$$\begin{array}{rcl} \hat{q}_1 & = & \Diamond\Box p, \\ \hat{q}_{m+1} & = & \Diamond(p \wedge \Diamond(\neg p \wedge \hat{q}_m)), \\ \hat{q}'_j & = & \hat{q}_j \wedge \neg\hat{q}_{j+1}, \\ \hat{r}_n & = & \Diamond\hat{q}'_n \wedge \Box(\hat{q}'_n \to \Diamond\Box\neg p), \end{array}$$

where $\hat{r}_1, \hat{r}_2, \hat{r}_3, \ldots$ must have the property that 'any atom of them (finite conjunction of $\hat{r}_i$'s and negations of $\hat{r}_i$'s) is satisfiable.' However, $\hat{r}_1$ is not **S4**-satisfiable; each of the formulas $\hat{r}_2, \hat{r}_3, \hat{r}_4, \ldots$ is **S4**-satisfiable, but the atom $\hat{r}_2 \wedge \neg\hat{r}_3 \wedge \hat{r}_4$ is not.

Further, it is assumed that the formulas $\hat{r}_1, \hat{r}_2, \hat{r}_3, \ldots$ are substituted for variables in some formula, denoted in [5] by ψ_A^{S4}, with its satisfiability (or non-satisfiability) being preserved. But the structure of ψ_A^{S4} is such that, even if we suppose that $\hat{r}_1, \hat{r}_2, \hat{r}_3, \ldots$ do have all of the required properties, one can show that in some cases ψ_A^{S4} is satisfiable, while its substitution instance is not. Note that our formula φ^* is a modification of ψ_A^{S4} (we add the conjunct $\neg q_{n+1}$ to a number of subformulas of ψ_A^{S4}).

Nevertheless, the general approach to the one-variable fragment of **S4** in [5] is sound. Here we show how to cure the defects of Halpern's construction.

So, let us consider **S4** and **Grz**. As in the case of **K4**

$$\begin{array}{rcl} \varphi \text{ is true} & \Longleftrightarrow & \varphi^* \text{ is } \mathbf{S4}\text{-satisfiable} \\ & \Longleftrightarrow & \varphi^* \text{ is } \mathbf{Grz}\text{-satisfiable.} \end{array}$$

For every m, let

$$\delta_1 = \Diamond\Box p, \quad \delta_{m+1} = \Diamond(p \wedge \Diamond(\neg p \wedge \delta_m)),$$

$$\alpha_m = \Box(p \to \Box(\neg p \wedge \delta_m \wedge \neg\delta_{m+1} \to \Box\Diamond p)).$$

As before, denote by φ^*_α the formula that is obtained from φ^* by substituting $\alpha_1, \ldots, \alpha_{2n+2}$ for $p_1, \ldots, p_n, q_0, \ldots, q_{n+1}$, respectively.

LEMMA 5 *The formula φ^*_α is constructed from φ^* by a polynomial time algorithm, and φ^*_α is* **S4***-satisfiable iff φ^*_α is* **Grz***-satisfiable iff φ^* is* **S4***-satisfiable iff φ^* is* **Grz***-satisfiable.*

The proof is similar to the proof of Lemma 1. The only difference is that when constructing M' we use the models M_m instead of the F_m; see Fig. 1 (the accessibility relation of M_m is reflexive and transitive).

Thus, we obtain

THEOREM 6 *The satisfiability (and so the decision) problem for the one-variable fragments of* **S4** *and* **Grz** *are* PSPACE*-complete.*

Now consider the Gödel–Löb provability logic **GL**. Besides the equivalences of Lemma 5 we also have

$$\varphi \text{ is true} \iff \varphi^* \text{ is } \mathbf{GL}\text{-satisfiable.}$$

As is well-known, **GL** is similar to **Grz**. So it is easy to modify the formulas $\alpha_1, \ldots, \alpha_{2n+2}$ for the case of **GL**. In fact, it is enough to put, for any m,

$$\delta_1 = \Diamond\Box^+ p, \quad \delta_{m+1} = \Diamond(p \wedge \Diamond(\neg p \wedge \delta_m)),$$

$$\alpha_m = \Box(p \to \Box(\neg p \wedge \delta_m \wedge \neg\delta_{m+1} \to \Box\Diamond^+ p)),$$

where $\Box^+\psi = \psi \wedge \Box\psi$, $\Diamond^+\psi = \psi \vee \Box\psi$.

An algorithm deciding the satisfiability problem for **GL** in polynomial space can be found in [3], see Theorem 18.29. Therefore, we obtain

THEOREM 7 *The satisfiability (and so the decision) problem for the one-variable fragment of* **GL** *is* PSPACE*-complete.*

Moreover, using the argument above one can prove the following more general

THEOREM 8 *Let $\mathbf{K} \subseteq L \subseteq \mathbf{Grz}$ or $\mathbf{K} \subseteq L \subseteq \mathbf{GL}$. Then the satisfiability (and so the decision) problem for the one-variable fragment of L is* PSPACE*-hard.*

3 Some related problems

3.1 NP-hardness

Theorem 8 does not solve all our problems for **GL**: there are infinitely many non-equivalent pairs of formulas in **GL**, for example, $\Box\bot$, $\Box\Box\bot$, $\Box\Box\Box\bot$, etc. Nevertheless, the variable-free fragment of **GL** is quite simple: we describe an algorithm deciding this fragment in polynomial time. We will use two well-known facts: (1) the variable-free fragment of **GL** is complete with respect to all finite linear **GL**-frames (see the description of $\mathfrak{F}^{<\infty}_{\mathbf{GL}}(0)$, i.e., universal frame of rank 0, in [3], p. 275) and (2) if φ is refuted in some **GL**-frame then φ is refuted in its subframe of height not exceeding the length of φ, see [1] or the proof of Theorem 18.13 in [3].

So, let φ be a variable-free formula of length k and let $\varphi_1, \ldots, \varphi_n = \varphi$ be all subformulas of φ such that if φ_i is a subformula of φ_j then $i \leq j$. We construct the table

	φ_1	φ_2	$\cdots$	φ_{n-1}	φ_n
0	f	t			
1	f	f			
$\cdots$					
$k-1$	f	f			

where $\{0, \ldots, k-1\}$ is supposed to be the set of points of a **GL**-frame and aRb iff $b < a$.

If $i \models \varphi_j$ then we write 't' in the (i, φ_j) square of the table; otherwise we write 'f' (for example, in the table above we have $\varphi_1 = \bot$, $\varphi_2 = \Box\bot$). It is not hard to see that this table can be constructed by a polynomial time algorithm in the length of φ.

When the construction is complete we check the last column: if it contains at least one 'f' then $\varphi \notin \mathbf{GL}$; otherwise $\varphi \in \mathbf{GL}$.

Therefore, we obtain

THEOREM 9 *The satisfiability and the decision problems for the variable-free fragment of* **GL** *are decidable in polynomial time.*

The algorithm presented in the proof of Theorem 9 is so simple, and it may seems it is possible to use the ideas of this algorithm to prove similar results for other logics with other restrictions on the number of variables. But in fact we cannot do too much. A first reasonable step in this direction is to consider the logic **GLLin** (= **GL.3**, i.e., the logic of all linear frames for **GL**): **GLLin** is the greatest logic among those having the same variable-free fragment as **GL** — and so the simplest of them. But such an attempt does not seem to be successful.

THEOREM 10 *The satisfiability problem for the one-variable fragment of* **GLLin** *is* NP*-complete.*

Indeed, the satisfiability problem for full **GLLin** (with countably infinite number of variables) is known to be in NP, see [3], Theorem 18.25, where **GLLin** is $\mathbf{GL} \oplus \boldsymbol{bw}_1$. So it is sufficient to reduce some NP-complete problem to the one-variable fragment of **GLLin**. We will reduce the satisfiability problem for Boolean formulas (in other words, the validity problem for formulas of the form $\exists p_1 \dots \exists p_n \varphi$, where φ is quantifier-free formula with the variables $p_1, \dots, p_n$).

Let φ be a Boolean formula with n variables. Without a loss of generality we will assume that the variables of φ are $p_1, \dots, p_n$. Replace every p_i in φ with $\Diamond(\Box^{i+1}\bot \wedge \Diamond^i \top \wedge p)$ and take the negation of the obtained formula. Denote the resulting formula by $\tilde{\varphi}$. It is easy to see that $\tilde{\varphi}$ is constructed in polynomial time and that

$$\varphi \text{ is satisfiable } \iff \tilde{\varphi} \notin \mathbf{GLLin}.$$

However, for some logics related to **GLLin** the algorithm of Theorem 9 can be modified. For example consider extensions of **S4**: **S4.3** and **Grz.3** (the first is determined by chains of clusters, while the second is characterised by finite linear frames with partial (i.e., reflexive) order).

THEOREM 11 *The one-variable fragments of* **S4.3** *and* **Grz.3** *are decidable in polynomial time.*

THEOREM 12 *The satisfiability problems for the two-variable fragments of* **S4.3** *and* **Grz.3** *are* NP*-complete.*

To prove Theorem 11 one can use the analigies of above-mentioned facts on **GL**: (1) the description of $\mathfrak{F}^{<\infty}_{\mathbf{S4.3}}(1)$ and $\mathfrak{F}^{<\infty}_{\mathbf{Grz.3}}(1)$ see in [3], p. 275, (2) the statement on linear approximability of these logics is contained in [3], Theorem 18.13. To prove Theorem 12 it is sufficient for any Boolean formula φ with n variables $p_1, \dots, p_n$ to consider the formula $\tilde{\varphi}$ obtained from φ by replacing every p_i with $\Diamond(\delta_i \wedge \neg\delta_{i+1} \wedge q)$, where δ_n is defined on p. 77.

3.2 Complexity function

The *complexity function* $f_L(n)$ for a logic (a set of formulas) L is defined as follows (see [3]):

$$f_L(n) = \max_{\substack{|\varphi| \,\leq\, n \\ \varphi \notin L}} \min_{\substack{F \models L \\ F \not\models \varphi}} |F|,$$

where $|F|$ is the number of points in F and $|\varphi|$ is the length of φ. For all PSPACE-hard logics L considered above, $f_L(n)$ cannot be bounded by any polynomial function. Indeed, take the formulas $\varphi_n = \forall p_1 \dots \forall p_n \bot$. It is easy to see that any model refuting φ_n^* must have at least $2^n + 1$ points. Since the length of the φ_n is a linear function of n, the length of the φ_n^* is a polynomial function of n, and so we can conclude that the complexity functions for **K**, **K4**, **S4**, **GL**, **Grz** cannot be bounded by any polynomial function (see [3] for a similar argument).

Given a logic L, denote by $L(m)$ the m-variable fragment of L. Then it follows from the discussion above that we have

THEOREM 13 *If* $L \in \{\mathbf{K}(0), \mathbf{K4}(0), \mathbf{S4}(1), \mathbf{GL}(1), \mathbf{Grz}(1)\}$ *then* $f_L(n)$ *cannot be bounded by any polynomial function.*

4 Acknowledgements

We are grateful to anonymous referees and our collegues, especially to Michael Zakharyaschev whose remarks and suggestions substantially helped to impove presentation.

BIBLIOGRAPHY

[1] S. N. Artemov. Modal logics axiomatizing provability. *Mathematics of the USSR, Izvestia*, 49:1123–1154, 1985.
[2] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic.* Cambridge University Press, 2001.
[3] A. Chagrov and M. Zakharyaschev. *Modal Logic.* Oxford University Press, 1997.
[4] M. R. Garey and D. S. Johnson. *Computers and Intractability, A Guide to the Theory of NP-completeness.* San Francisco, 1979.
[5] J. Y. Halpern. The effect of bounding the number of primitive propositions and the depth of nesting on the complexity of modal logic. *Artificial Intelligence*, 75(2):361–372, 1995.
[6] R.E. Ladner. The computational complexity of provability in systems of modal logic. *SIAM Journal on Computing*, 6:467–480, 1977.
[7] W. J. Savitch. Relationships between nondeterministic and deterministic tape complexity. *Journal of Computer and System Science*, 4:177–192, 1970.
[8] E. Spaan. *Complexity of Modal Logics.* PhD thesis, University of Amsterdam, Department of Mathematics and Computer Science, 1993.
[9] L. Stockmeyer. Classifying the computational complexity of problems. *The Journal of Symbolic Logic*, 52(1):1–43, 1987.
[10] M. Zakharyaschev, F. Wolter, and A. Chagrov. Advanced modal logic. In D. M. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic*, volume 3, pages 83–266. Kluwer Academic Publishers, 2nd edition, 2001.

A. V. Chagrov
Department of Mathematics
Tver State University
Zhelyabova Street, 33
Tver, 170000, Russia
E-mail: chagrov@mail.ru

M. N. Rybakov
Department of Mathematics
Tver State University
Zhelyabova Street, 33
Tver, 170000, Russia
E-mail: m_ rybakov@mail.ru, s000125@tversu.ru