On PSPACE-decidability in Transitive Modal Logics

I. Shapirovsky

ABSTRACT. In this paper we describe a new method, allowing us to prove PSPACE-decidability for transitive modal logics. We apply it to $\mathbf{L}_1$ and $\mathbf{L}_2$, modal logics of Minkowski spacetime. We also show how to extend this method to some other transitive logics.

1 Introduction

Computational complexity of modal logics was first studied by Ladner in [11]. To obtain upper complexity bounds, he modified the *tableau method* from [10][1]. Later various tableau-based methods were used in PSPACE-decidability proofs for a number of monomodal logics (like $\mathbf{K}, \mathbf{K4}, \mathbf{S4}$, etc. [11],[14]), and also for multimodal and tense logics, cf. [8],[15].

In this paper we propose an alternative proof for PSPACE upper bounds in transitive modal logics. The satisfiability problem is reduced to satisfiability in some "standard" finite frames. To obtain these frames, we apply *selective filtration* (see e.g. [4]) and extract a finite submodel from the canonical model. The height of this submodel is polynomially bounded, due to the *maximality property* [6] of the canonical model.

This construction allows us to give a rather simple description of the decision procedure. The method happens to be "robust" – after adding extra axioms (such as density or McKinsey axiom), only a slight modification is sufficient.

To illustrate our method, we consider two particular logics, $\mathbf{L}_1$ and $\mathbf{L}_2$. These logics were introduced in the study of chronological future modalities in Minkowski spacetime [7],[12]. In [12] the finite model property (FMP) of these logics was proved. "Standard" finite frames for $\mathbf{L}_1$ and $\mathbf{L}_2$ can be obtained following the lines of [12]. Basing on this construction, we describe the deciding deterministic algorithm working within a polynomial space. It follows that $\mathbf{L}_1$ and $\mathbf{L}_2$ are PSPACE-complete (the lower bounds can be obtained by Ladner's reduction of the QBF-validity problem to the modal satisfiability problem [11]).

[1]The tableau method for the propositional calculi was first developed in [1].

We show how to apply our method to some other transitive logics. In particular, for the logic **K4** and its extensions by density, reflexivity, confluence, McKinsey axiom, we propose a new proof of PSPACE-decidability.

2 Preliminaries

In this paper we consider propositional normal monomodal logics containing **K4**.

We assume that $\Diamond$, $\rightarrow$, $\bot$ are the basic connectives, and $\Box$, $\neg$, $\vee$, $\wedge$, $\top$ are derived. PV denotes the countable set of propositional variables. For a modal logic Λ and a modal formula φ, the notation $\Lambda + \varphi$ denotes the smallest modal logic containing $\Lambda \cup \{\varphi\}$; $\Lambda \vdash \varphi$ means $\varphi \in \Lambda$. $Sub(\varphi)$ denotes the set of all subformulas of φ, $PV(\varphi) := PV \cap Sub(\varphi)$. Here are the names for some particular axioms:

$$
\begin{array}{ll}
A4 := \Diamond\Diamond p \rightarrow \Diamond p & \textit{transitivity}, \\
AT := p \rightarrow \Diamond p & \textit{reflexivity}, \\
AD := \Diamond\top & \textit{seriality}, \\
A1 := \Box\Diamond p \rightarrow \Diamond\Box p & \textit{McKinsey axiom}, \\
A2 := \Diamond\Box p \rightarrow \Box\Diamond p & \textit{confluence}, \\
Ad = Ad_1 := \Diamond p \rightarrow \Diamond\Diamond p & \textit{density}, \\
Ad_2 := \Diamond p_1 \wedge \Diamond p_2 \rightarrow \Diamond(\Diamond p_1 \wedge \Diamond p_2) & \textit{2-density};
\end{array}
$$

and the names for some logics:

$$
\begin{array}{lll}
\mathbf{K4} := \mathbf{K} + A4, & \mathbf{K4d} := \mathbf{K4} + Ad, & \mathbf{S4} := \mathbf{K4} + AT, \\
\mathbf{L}_1 := \mathbf{K4} + AD + Ad_2, & \mathbf{L}_2 := \mathbf{L}_1 + A2. &
\end{array}
$$

For a logic Λ let $\Lambda.1 := \Lambda + A1$, $\Lambda.2 := \Lambda + A2$.

As usual, a *(Kripke) frame* is a pair (W, R), where $W \neq \varnothing$, $R \subseteq W \times W$. We consider only transitive frames. A *(Kripke) model* is a Kripke frame with a valuation: $M = (W, R, \theta)$, where $\theta : PV \longrightarrow 2^W$, 2^W denotes the power set of W. For a model $M = (W, R, \theta)$ or a frame $F = (W, R)$, the notation $x \in M$ or $x \in F$ means $x \in W$. As usual, for $x \in W$, $V \subseteq W$ let $R(x) := \{y \mid xRy\}$, $R(V) := \bigcup_{x \in V} R(x)$, $R|V := R \cap (V \times V)$. We also put $W^x := \{x\} \cup R(x)$, $F^x := (W^x, R|W^x)$.

A model $M_1 = (W_1, R_1, \theta_1)$ is a *(weak) submodel* of $M = (W, R, \theta)$ (notation: $M_1 \subseteq M$) if

$$W_1 \subseteq W,\ R_1 \subseteq R,\ \theta_1(p) = \theta(p) \cap 2^{W_1}$$

for every $p \in PV$. If $R_1 = R|W_1$, then M_1 is called the *restriction* of M to W_1 and denoted by $M|W_1$. The submodel $M^x := M|W^x$ is called a *cone* in M.

The sign $\vDash$ denotes the truth at a point of a model and also the validity in a frame. For a class of frames $\mathcal{F}$, $\mathbf{L}(\mathcal{F})$ denotes the set of all formulas that are valid in all frames from $\mathcal{F}$. For a single frame F, $\mathbf{L}(F)$ abbreviates $\mathbf{L}(\{F\})$. For a logic Λ, if $\mathbf{L}(F) \supseteq \Lambda$, then we say that F is Λ-*frame*. Recall that Λ is *Kripke-complete* if $\Lambda = \mathbf{L}(\mathcal{F})$ for some class of frames $\mathcal{F}$.

A formula φ is *satisfiable in a model* M if for some $x \in M$ we have $M, x \vDash \varphi$; φ is *satisfiable in a frame* F if φ is satisfiable in some model over F. For a class of frames $\mathcal{F}$, φ is $\mathcal{F}$-*satisfiable* if φ is satisfiable in some $F \in \mathcal{F}$. φ is Λ-*satisfiable* if φ is satisfiable in some Λ-frame. Note that if Λ is Kripke-complete, then we have: φ is Λ-satisfiable $\Leftrightarrow \Lambda \nvdash \neg\varphi$.

The *disjoint union* $F_1 \sqcup F_2$ and the *ordinal sum* $F_1 + F_2$ of frames $F_1,\ F_2$ are defined in a standard way. The notation $f : F_1 \twoheadrightarrow F_2$ means that f is a p-morphism from F_1 onto F_2, and $F_1 \twoheadrightarrow F_2$ means that $f : F_1 \twoheadrightarrow F_2$ for some f.

Recall that a *cluster* in (W, R) is an equivalence class under the relation $\sim_R := (R \cap R^{-1}) \cup Id_W$, where Id_W is the equality relation on W. For a point x, $\overline{x}$ denotes its cluster. C_0 denotes a *degenerate cluster*, i.e. an irreflexive singleton; C_1 denotes a reflexive singleton; C_n denotes an n-element cluster for $n \geq 2$. Let $W/{\sim_R} := \{\overline{x} \mid x \in W\}$. For clusters $C, D \in W/{\sim_R}$ we put

$$C \leqq_R D := D \subseteq R(C),\ C <_R D := C \leqq_R D \text{ and } C \neq D.$$

Note that the relations $\leqq_R, <_R$ are transitive and antisymmetric, $<_R$ is irreflexive, and $C \leqq C$ iff C is non-degenerate. A cluster D is a *successor* of C, if $C <_R D$ and there is no cluster C' such that $C <_R C' <_R D$. For $x, y \in F$ we say that y is a *successor* of x, if $\overline{y}$ is a successor of $\overline{x}$. The frame $F/{\sim_R} := (W/{\sim_R}, \leqq_R)$ is called the *skeleton* of F (and of every model over F). A point $x \in F$ is called *maximal* (*minimal*) if its cluster is maximal (minimal) in $F/{\sim_R}$.

In this paper a frame (W, R) is called *rooted* if for some x $W = W^x$, and the cluster $\overline{x}$ is one-element. A *tree* is a rooted frame (W, R) such that R is transitive and antisymmetric, and $R^{-1}(x)$ is a chain for every $x \in W$. A frame F is called a *quasitree* if its skeleton $F/{\sim_R}$ is a tree.

Let us recall the notion of *selective filtration* [12], cf. [2],[4].

DEFINITION 1. Let M be a Kripke model, Ψ a set of formulas closed under subformulas. A submodel $M_1 \subseteq M$ (with the relation R_1) is called a *selective filtration of M through* Ψ (notation: $M_1 \in \mathcal{SF}(M, \Psi)$), if for any

$x \in M_1$, for any formula φ

$$\Diamond\varphi \in \Psi \ \&\ M, x \vDash \Diamond\varphi \Rightarrow \exists y \in R_1(x)\ M, y \vDash \varphi.$$

The following lemma is proved easily by induction on the length of a formula φ.

LEMMA 2. *If $M_1 \in \mathcal{SF}(M, \Psi)$, then for any $x \in M_1$, for any $\varphi \in \Psi$*

$$M, x \vDash \varphi \Leftrightarrow M_1, x \vDash \varphi.$$

The following lemma states the "maximality property" of a canonical model, cf. [6],[12].

LEMMA 3. *Let $\mathfrak{M}$ be the canonical model of a logic Λ, and assume that $\mathfrak{M}, x \vDash \varphi$. Consider the set of all those clusters in $\mathfrak{M}^x$, in which φ is satisfied:*[2]

$$\Gamma := \{C \subseteq \mathfrak{M}^x \mid \exists y \in C\ \varphi \in y\}.$$

Then the model $\mathfrak{M}|\bigcup\Gamma$ contains a maximal cluster.

3 Decrease of thickness and branching

Let $|V|$ denote the cardinality of a set V. Consider a finite frame $F = (W, R)$. For a cluster C let $next(C)$ denote the set of all successors of C, $\mathbf{b}(C) := |next(C)|$. We put:

$$\begin{array}{ll} \mathbf{h}(F) := \max\{\ |\Sigma|\ \mid \Sigma \text{ is a } <_R\text{-chain in } W/{\sim_R}\} & \textit{height}, \\ \mathbf{b}(F) := \max\{\mathbf{b}(C) \mid C \in W/{\sim_R}\} & \textit{branching}, \\ \mathbf{t}(F) := \max\{\ |C|\ \mid C \in W/{\sim_R}\} & \textit{thickness}. \end{array}$$

Note that $\mathbf{h}(F) = \mathbf{h}(F/{\sim_R}),\ \mathbf{b}(F) = \mathbf{b}(F/{\sim_R})$.
For a model M over F we put

$$\mathbf{h}(M) := \mathbf{h}(F),\ \mathbf{b}(M) := \mathbf{b}(F),\ \mathbf{t}(M) := \mathbf{t}(F).$$

Two following simple lemmas allow us to decrease the thickness and the branching of a given model.

LEMMA 4. *Assume that $M, y \vDash \varphi$, $n = |Sub(\varphi)|$. Then there exists a restriction M' of M such that $M', y \vDash \varphi$, $\mathbf{t}(M') \leq n$, and the skeletons of M, M' are isomorphic.*

[2] Recall that in the canonical model $\mathfrak{M}, y \vDash \varphi$ iff $\varphi \in y$.

Proof. For a cluster C let

$$\Psi(C) := \{\psi \in Sub(\varphi) \mid \exists x \in C\ M, x \vDash \psi\}.$$

Now define the set V_C as follows.

If $|C| \leq n$, then $V_C := C$; if $|C| > n$, then for every $\psi \in \Psi(C)$ we choose a point x_ψ such that $M, x \vDash \psi$ (in the particular case, when $C = \overline{y}$, we put $x_\varphi := y$), and let $V_C := \{x_\psi \mid \psi \in \Psi(C)\}$.

We put $W' := \bigcup_{C \in F/\sim_R} V_C$, $M' := M|W'$. Obviously, $\mathbf{t}(M') \leq n$ and the skeletons of M and M' are isomorphic.

It is easy to see that $M' \in \mathcal{SF}(M, Sub(\varphi))$, so $M', y \vDash \varphi$. ■

LEMMA 5. *Let M be a finite*[3] *model over a quasitree, $M, y \vDash \varphi$, $n = |Sub(\varphi)|$. Then there exists a restriction M' of M such that $M', y \vDash \varphi$, $\mathbf{b}(M') \leq n$.*

Proof. It is sufficient to consider the case $M = M^y$.

Induction on the number of clusters such that $\mathbf{b}(C) > n$.

For the induction step, suppose $\mathbf{b}(C) > n$ for some cluster C. Let

$$\Phi(C) := \{\Diamond\psi \in Sub(\varphi) | \text{ for some } \overline{z} \in next(C)\ M, z \vDash \psi \vee \Diamond\psi\}.$$

Suppose that $\Phi(C) := \{\Diamond\psi_1, \ldots, \Diamond\psi_k\}$. For every formula $\Diamond\psi_i$ we choose $\overline{x}_i \in next(C)$ such that ψ_i is satisfiable in M^{x_i}. If $\overline{z} \in next(C) - \{\overline{x}_1, \ldots, \overline{x}_k\}$, then we say that the cone M^z is *redundant* for C.

Let M_1 be the restriction of M, obtained after elimination of all cones redundant for C. One can see that $M_1 \in \mathcal{SF}(M, Sub(\varphi))$, thus $M_1, y \vDash \varphi$. By the induction hypothesis, there exists a restriction M' of M_1 such that $M', y \vDash \varphi$, $\mathbf{b}(M') \leq n$. ■

4 Completeness results for $\mathbf{L}_1$ and $\mathbf{L}_2$

The following lemmas are proved rather easily, cf. [7],[12].

LEMMA 6. *For all n*

$$\mathbf{K4} + Ad_2 \vdash \Diamond p_1 \wedge \ldots \wedge \Diamond p_n \to \Diamond(\Diamond p_1 \wedge \ldots \wedge \Diamond p_n).$$

LEMMA 7.

(i) $F \vDash Ad_2$ iff F is 2-dense*, i.e.,*

$$\forall x \forall y_1 \forall y_2 (xRy_1\ \&\ xRy_2 \to \exists z(xRz\ \&\ zRy_1\ \&\ zRy_2)).$$

[3] We assume that M is finite only for the sake of simplicity.

(ii) A finite frame F is 2-dense iff every degenerate non-maximal $C \in F/{\sim_R}$ has a unique successor D, and D is non-degenerate.

By Sahlqvist's Theorem we obtain

LEMMA 8. *The logics $\mathbf{L}_1, \mathbf{L}_2$ are canonical.*

Recall that the chronological future relation $\prec$ in Minkowski spacetime $\mathbb{R}^n,\ n \geq 2$ is defined as follows:

$$(x_1, \ldots, x_n) \prec (y_1, \ldots, y_n) \Leftrightarrow \sum_{i=1}^{n-1} (x_i - y_i)^2 < (x_n - y_n)^2 \ \&\ x_n < y_n.$$

Let us quote the main completeness results for the logics $\mathbf{L}_1$ and $\mathbf{L}_2$ [12].

THEOREM 9. $\mathbf{L}(\mathbb{R}^n, \prec) = \mathbf{L}_2,\ n \geq 2.$

THEOREM 10. *Let X be an open connected domain in $\mathbb{R}^2$ bounded by a closed smooth curve. Then $\mathbf{L}(X, \prec) = \mathbf{L}_1$.*

These logics can also be interpreted as fragments of the interval logic of the real line. Let I be the set of all open intervals on $\mathbb{R}$:

$$\mathrm{I} := \{\ ]a, b[\subseteq \mathbb{R} \mid a < b\}.$$

Consider the following relation between intervals:

$$]a_1, b_1[\ \sqsubset\ ]a_2, b_2[\ := a_2 < a_1 \& b_1 < b_2,$$

and its converse $\sqsupset$.

THEOREM 11. $\mathbf{L}(\mathrm{I}, \sqsubset) = \mathbf{L}_2$, $\mathbf{L}(\mathrm{I}, \sqsupset) = \mathbf{L}_1$.

5 Strong finite model property of $\mathbf{L}_1,\ \mathbf{L}_2$

In this section we show how to reduce the $\mathbf{L}_1$- and $\mathbf{L}_2$-satisfiability of a given formula to satisfiability in appropriate finite frames.

Let $\mathcal{F}_1$ be the class of all finite $\mathbf{L}_1$-frames,

$$\mathcal{F}_2 := \{F + C \mid F \in \mathcal{F}_1,\ C \text{ is a finite non-degenerate cluster}\},$$

and let μ_i be the class of Kripke models over frames from $\mathcal{F}_i,\ i = 1, 2$.

For 2-dense frames it is convenient to modify the function $\mathbf{h}$. Namely, for a $<_R$-chain Σ in $W/{\sim_R}$ let $\mathbf{h}_r(\Sigma)$ be the number of all non-degenerate clusters in Σ. The *r-height* of F (and of a model over F) is defined as follows:

$$\mathbf{h}_r(F) := \max\{\mathbf{h}_r(\Sigma) \mid \Sigma \text{ is a } <_R \text{-chain in } W/{\sim_R}\}$$

In [12] it was proved that the logics $\mathbf{L}_1$ and $\mathbf{L}_2$ have the FMP. This proof actually yields the following

LEMMA 12. *Consider a formula φ, $n = |Sub(\varphi)|$.*

(i) If φ is $\mathbf{L}_1$-satisfiable, then φ is satisfiable in a frame $F \in \mathcal{F}_1$ such that $\mathbf{h}_r(F) \leq n$.

(ii) If φ is $\mathbf{L}_2$-satisfiable then φ is satisfiable in a frame $F \in \mathcal{F}_2$ such that $\mathbf{h}_r(F) \leq n + 1$.

Proof.

(i) Let $\mathfrak{M}$ be the canonical model of $\mathbf{L}_1$ with the accessibility relation R. For some x_0 we have $\mathfrak{M}, x_0 \vDash \varphi$. We will construct a model $M \subseteq \mathfrak{M}$ such that $M \in \mu_1,\ \mathbf{h}_r(M) \leq n,\ M, x_0 \vDash \varphi$.

Let $\Phi := Sub(\varphi) \cup \{\Diamond\top\}$. For every $x \in \mathfrak{M}$ we put:

$$\Phi_x := \{\Diamond\psi \mid \Diamond\psi \in \Phi \cap x\}, \quad \phi_x := \bigwedge_{\Diamond\psi \in \Phi_x} \Diamond\psi,$$

$$\Phi_x^{\sim} := \{\Diamond\psi \in \Phi_x \mid \exists t \sim_R x \ \ \psi \in t\},\ \Phi_x^{\uparrow} := \Phi_x - \Phi_x^{\sim},$$

$$Y_x := \{y \mid xRy,\ \phi_x \in y\}.$$

Due to the seriality, $\phi_x \in x$, and by Lemma 6, $\Diamond\phi_x \in x$. Thus Y_x is non-empty, and by Lemma 3 Y_x contains a maximal point.

For every $x \in \mathfrak{M}$ we choose a point x', which is maximal in Y_x (we put $x' := x$ if x already is maximal in Y_x). It is easy to see that x' is reflexive. Indeed, $\Diamond\phi_x \in x'$, thus for some $y \in R(x')$ we have $y \in Y_x$. Since x' is maximal in Y_x, we have $y \in \overline{x'}$, and so $\overline{x'}$ is non-degenerate, i.e. x' is reflexive. Note that $\Phi_x = \Phi_{x'}$ and for every $z \in \mathfrak{M}$ if $\overline{x'} <_R \overline{z}$ then $|\Phi_z| < |\Phi_x|$.

Now by induction we construct a filtration $M \in \mathcal{SF}(\mathfrak{M}, \Psi)$. We also define an auxiliary set X_k at every stage k.

Stage 0. We put

$$W_0 := \{x_0, x_0'\},\ R_0 := \{(x_0, x_0'), (x_0', x_0')\}, X_0 := \{x_0'\}.$$

Let M_0 be the submodel of $\mathfrak{M}$ over the frame (W_0, R_0).

Stage k+1. Assume that on stage k we have a model M_k over a frame (W_k, R_k) such that $M_k \subseteq \mathfrak{M},\ M_k \in \mu_1,\ X_k \neq \varnothing$ and the following holds:

(1) if $x' \in W_k - X_k$, $\Diamond\psi \in \Phi_x$, then $\psi \in y$ for some $y \in R_k(x)$;

(2) if $x \in X_k$, then $|\Phi_x| \leq n - k$;

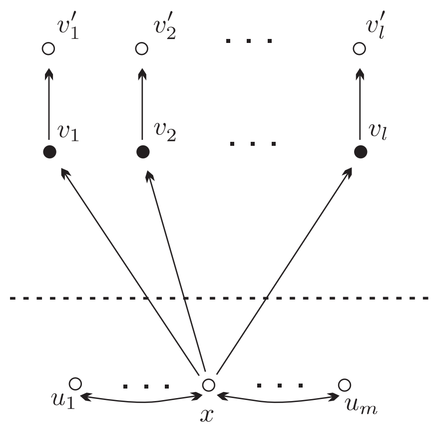


Figure 1.

(3) if $\overline{x}$ is non-degenerate in W_k and $\overline{x} <_R \overline{y}$, then $|\Psi_y| < |\Psi_x|$.

Now let us construct $M_{k+1},\ X_{k+1}$.

For every $x \in X_k$ we define the sets of points $U_x^\sim,\ U_x^\uparrow$ and U'_x as follows. If $\Phi_x^\sim = \{\Diamond\psi_1, \ldots, \Diamond\psi_m\}$, then there exist points $u_1, \ldots, u_m$ such that $u_i \ni \psi_i,\ u_i \sim_R x$. We put $U_x^\sim := \{u_1, \ldots, u_m\}$ (note that $\Phi_x^\sim \neq \varnothing$, since $\Diamond\top \in \Phi_x$). For $\Phi_x^\uparrow = \{\Diamond\chi_1, \ldots, \Diamond\chi_l\}$ we put $U_x^\uparrow := \{v_1, \ldots, v_l\}$ and $U'_x := \{v'_1, \ldots, v'_l\}$, where $v_i \ni \chi_i,\ \overline{x} <_R \overline{v_i}$ (if $\Phi_x^\uparrow = \varnothing$, we put $U_x^\uparrow := U'_x := \varnothing$), Figure 1. Let

$$R_x := \bigcup_{1\le i\le l} \{(x, v_i), (v_i, v'_i), (v'_i, v'_i)\} \ \cup\ (U_x^\sim \cup \{x\})^2;$$

$$W_{k+1} := \bigcup_{x\in X_k} (U_x^\sim \cup U_x^\uparrow \cup U'_x) \cup W_k, \quad X_{k+1} := \bigcup_{x\in X_k} U'_x.$$

Let R_{k+1} be the transitive closure of $R_k \cup \bigcup\limits_{x\in X_k} R_x$.

One can see that $M_{k+1} \subseteq \mathfrak{M},\ M_{k+1} \in \mu_1$. The property (1) holds due to the construction. The property (2) holds, since $|\Phi_y| < |\Phi_x|$ for any $x \in X_k,\ y \in U'_x$. If $\overline{x}$ is non-degenerate in M_{k+1}, then $\overline{x}$ contains a point of some X_i, $1 \le i \le k+1$, so the property (3) holds.

Due to the property (2), it follows that $X_{k+1} = \varnothing$ at some stage k. The construction terminates at this stage, and we put $M := M_{k+1}$. Due to the property (1), $M \in \mathcal{SF}(\mathfrak{M}, \Phi)$, so $M, x_0 \vDash \varphi$.

For every $x \in M$ $|\Psi_x| \leq n, |\Psi_x| \geq 1$, so by the property (3), we obtain $\mathbf{h}_r(M) \leq n$.

(ii) Let $\mathfrak{M}$ be the canonical model of $\mathbf{L}_2$, $\mathfrak{M}, x_0 \vDash \varphi$. As well as in (i), we construct a finite submodel M of $\mathfrak{M}$ such that $M \in \mu_1$, $M, x_0 \vDash \varphi$, and $\mathbf{h}_r(M) \leq n$.

Since $\mathfrak{M}$ is serial, $\mathfrak{M}^{x_0} = \{y | y \in \mathfrak{M}^{x_0} \& \Diamond\top \in y\}$. By Lemma 3, $\mathfrak{M}^{x_0}$ contains a maximal cluster C. By confluence and seriality, C is the non-degenerate final cluster in $\mathfrak{M}^{x_0}$.

If M contains points from C, then the frame F of M is confluent. Let C' be the copy of the final cluster of F, $F' := F + C'$. Obviously, $F' \in \mathcal{F}_2$, $\mathbf{h}_r(F) \leq n+1$. Since $F' \twoheadrightarrow F$, φ is satisfiable in F.

Assume that M does not contain points from C. Consider the set of formulas $\Psi = \{\psi \in Sub(\varphi) \mid \exists x \in C\ \psi \in x\} \cup \{\top\}$. For $\Psi = \{\psi_1, \ldots, \psi_k\}$ we put $C' := \{x_1, \ldots, x_k\}$, where $\mathfrak{M}, x_i \vDash \psi_i$, $x_i \in C$. Then the submodel $M' \subseteq \mathfrak{M}$ obtained by putting the cluster C' on the top of M is in μ_2. One can see that $M', x_0 \vDash \varphi$ and $\mathbf{h}_r(M') \leq n+1$. ∎

Let $\mathcal{G}_1$ be the class of all quasitrees from $\mathcal{F}_1$,

$$\mathcal{G}_2 := \{G + C \mid G \in \mathcal{G}_1,\ C \text{ is a finite non-degenerate cluster}\},$$
$$\mathcal{G}_1(n) := \{G \in \mathcal{G}_1 \mid \mathbf{h}_r(G) \leq n,\ \mathbf{b}(G) \leq n,\ \mathbf{t}(G) \leq n\},$$
$$\mathcal{G}_2(n) := \{G \in \mathcal{G}_2 \mid \mathbf{h}_r(G) \leq n+1,\ \mathbf{b}(G) \leq n,\ \mathbf{t}(G) \leq n\}.$$

Let us formulate the following *strong finite model property* (SFMP) of $\mathbf{L}_1$ and $\mathbf{L}_2$.

LEMMA 13. *Consider a formula φ, $n = |Sub(\varphi)|$.*

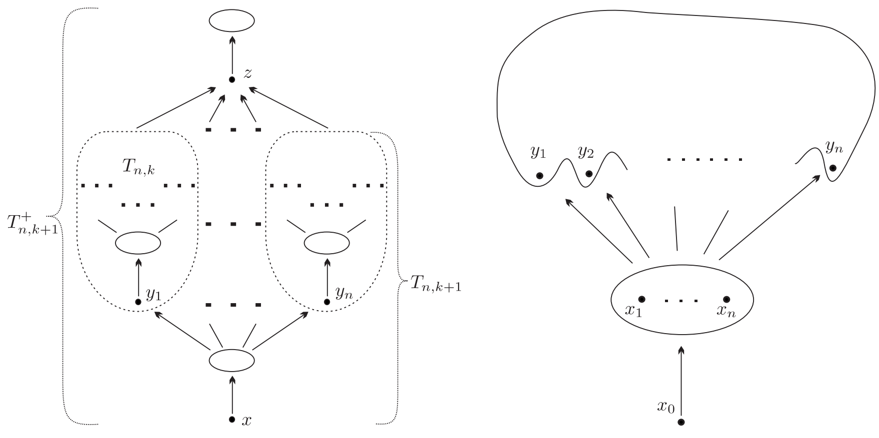
(i) φ is $\mathbf{L}_1$-satisfiable $\Leftrightarrow$ φ is $\mathcal{G}_1(n)$-satisfiable.

(ii) φ is $\mathbf{L}_2$-satisfiable $\Leftrightarrow$ φ is $\mathcal{G}_2(n)$-satisfiable.

Proof.

(i) Suppose φ is $\mathbf{L}_1$-satisfiable. By Lemma 12, φ is satisfiable in some frame $F \in \mathcal{F}_1$ such that $\mathbf{h}_r(F) \leq n$. Obviously, we can assume that F has the initial cluster (in which φ is satisfied). By standard unravelling argument, F is a p-morphic image of some quasitree $F' \in \mathcal{G}_1$, and $\mathbf{h}_r(F) = \mathbf{h}_r(F')$ (see [12] for more details). By the p-morphism lemma φ is satisfiable in F', and by Lemma 4, φ is satisfiable in some $F'' \in \mathcal{G}_1$ such that $\mathbf{t}(F'') \leq n$, $\mathbf{h}_r(F'') \leq n$.

To decrease the branching, we proceed in the same way as in Lemma 5. Note that the transformation described in Lemma 5 preserves 2-density:



(**a**) (**b**)

Figure 2.

every degenerate cluster still has at most one non-degenerate successor. The following slight modification allows us to preserve seriality: if for some cluster C we have $\mathbf{b}(C) > n$ and $\Phi(C) = \varnothing$ (in notation of Lemma 5), then we put $\Phi(C) := \{\Diamond\top\}$.

By applying this transformation to F'' we obtain $\mathbf{L}_1$-quasitree G such that $\mathbf{b}(G) \leq n$ and φ is satisfiable in G. Obviously, $\mathbf{h}_r(G) \leq \mathbf{h}_r(F'')$, $\mathbf{t}(G) \leq \mathbf{t}(F'')$, thus $G \in \mathcal{G}_1(n)$.

(ii) Suppose φ is $\mathbf{L}_2$-satisfiable. By Lemma 12, φ is satisfiable in some frame $F^+ \in \mathcal{F}_2$ such that $\mathbf{h}_r(F^+) \leq n+1$, i.e. $F^+ = F + C$, where $F \in \mathcal{G}_1(n)$, C is a non-degenerate cluster. By Lemma 4, we can assume that $|C| \leq n$.

As well as in (i), we modify F into $G \in \mathcal{G}_1(n)$. Then $G + C \in \mathcal{G}_2(n)$. It is not difficult to check that φ is satisfiable in $G + C$. ■

Actually, this lemma is sufficient to show that the logics $\mathbf{L}_1$ and $\mathbf{L}_2$ are PSPACE-decidable. Indeed, it is possible to describe an algorithm checking satisfiability in all frames from $\mathcal{G}_1(n)$ (or from $\mathcal{G}_2(n)$) within the space polynomial of n. However, to simplify the algorithm, we can only check a single frame, as explained below.

Let $\mathcal{T}_{n,1}$ be the class of all frames isomorphic to $C_0 + C_n$. We put

$$\mathcal{T}_{n,k+1} := \{F + (G_1 \sqcup \ldots \sqcup G_n) \mid F \in \mathcal{T}_{n,1},\ G_1, \ldots, G_n \in \mathcal{T}_{n,k}\},$$

$$\mathcal{T}^+_{n,k} := \{F + F' \mid F \in \mathcal{T}_{n,k},\ F' \in \mathcal{T}_{n,1}\}.$$

Let $T_{n,k} \in \mathcal{T}_{n,k},\ T^+_{n,k} \in \mathcal{T}^+_{n,k}$ (Figure 2a).

LEMMA 14. *Consider a formula φ, $n = |Sub(\varphi)|$.*

(i) φ is $\mathbf{L}_1$-satisfiable $\Leftrightarrow$ φ is satisfiable at the root of $T_{n,n}$.

(ii) φ is $\mathbf{L}_2$-satisfiable $\Leftrightarrow$ φ is satisfiable at the root of $T^+_{n,n}$.

Proof.

(i) $(\Rightarrow)$. By induction on r-height it is easy to check that $T_{n,n} \twoheadrightarrow G$ for every $G \in \mathcal{G}_1(n)$. The statement follows from Lemma 13 and the p-morphism lemma.
$(\Leftarrow)$. Note that $T_{n,n}$ is $\mathbf{L}_1$-frame.

(ii) Similar to (i). ■

6 PSPACE-completeness for $\mathbf{L}_1$ and $\mathbf{L}_2$

In this section we prove the PSPACE-completeness for the logics $\mathbf{L}_1$ and $\mathbf{L}_2$.

First we show that $\mathbf{L}_1, \mathbf{L}_2 \in$ PSPACE. By Lemma 14, it is sufficient to describe the algorithms deciding whether a given formula is satisfiable at the roots of $T_{n,n}$ and $T^+_{n,n}$, using space polynomial of n.

Consider a formula φ and assume that $Sub(\varphi) = \{\psi_1, \dots, \psi_n\}$, $PV(\varphi)$=$\{p_1, \dots, p_m\}$. Let us order $Sub(\varphi)$ as follows: for $i \leq m$ we put $\psi_i := p_i$, and if ψ_i is a subformula of ψ_j then $i \leq j$. We can achieve this within $O(n \log n)$ units of space by making an array of pointers. Note that $\psi_n = \varphi$.

Consider a boolean vector $\mathbf{v} = (\mathrm{v}_1, \dots, \mathrm{v}_n) \in \{0,1\}^n$. We put $\varphi^{\mathbf{v}} := \bigwedge\limits_i \psi_i^{\mathrm{v}_i}$, where $\psi^0 := \neg\psi,\ \psi^1 := \psi$.

DEFINITION 15. A boolean vector $\mathbf{v} \in \{0,1\}^n$ is called *φ-consistent* in a frame F at a point x (notation: $F, x \Vdash_\varphi \mathbf{v}$) if for some valuation θ we have $F, \theta, x \models \varphi^{\mathbf{v}}$. Boolean vectors $\mathbf{v}^1, \dots, \mathbf{v}^l \in \{0,1\}^n$ are called *simultaneously φ-consistent* in F on a tuple $\mathbf{y} = (y_1, \dots, y_l) \in W^l$ (notation: $F, \mathbf{y} \Vdash_\varphi (\mathbf{v}^1, \dots, \mathbf{v}^l)$) if for some valuation θ for all $i = 1 \dots l$ we have $F, \theta, y_i \models \varphi^{\mathbf{v}^i}$.

If x is the root of F and y is the root of G, then $F \Vdash_\varphi \mathbf{v}$ abbreviates $F, x \Vdash_\varphi \mathbf{v}$ and $F + G \Vdash_\varphi (\mathbf{v}, \mathbf{u})$ abbreviates $F + G, (x, y) \Vdash_\varphi (\mathbf{v}, \mathbf{u})$.

Let us reformulate Lemma 14.

LEMMA 16.

(i) φ *is* $\mathbf{L}_1$*-satisfiable* $\Leftrightarrow$ *there exists* $\mathbf{v} \in \{0,1\}^n$ *such that* $T_{n,n} \Vdash_\varphi \mathbf{v}$ *and* $\mathrm{v}_n = 1$.

(ii) φ *is* $\mathbf{L}_2$*-satisfiable* $\Leftrightarrow$ *there exist* $\mathbf{v}, \mathbf{u} \in \{0,1\}^n$ *such that* $T^+_{n,n} \Vdash_\varphi (\mathbf{v}, \mathbf{u})$ *and* $\mathrm{v}_n = 1$.

Consider the frame $F = T_{n,1} + G$, where G has exactly n minimal clusters $\overline{y}_1, \ldots, \overline{y}_n$, and let x_0 be the root of $T_{n,1}$ (Figure 2b). The truth value of a formula φ at x_0 in a model over F is fully determined by the truth values of its variables in $T_{n,1}$ and the truth values of its subformulas at $y_1, \ldots, y_n$. In Appendix we describe the algorithm *SatLoc* working within a space polynomial of n, deciding whether $F, (x, y_1, \ldots, y_n) \Vdash_\varphi (\mathbf{v}, \mathbf{v}^1, \ldots, \mathbf{v}^n)$, provided $F, (y_1, \ldots, y_n) \Vdash_\varphi (\mathbf{v}^1, \ldots, \mathbf{v}^n)$.

LEMMA 17. *Let* $\mathbf{v}, \mathbf{u} \in \{0,1\}^n$.

(i) $T_{n,k+1} \Vdash_\varphi \mathbf{v} \Leftrightarrow$ *there exist* $\mathbf{v}^1, \ldots, \mathbf{v}^n \in \{0,1\}^n$ *such that* $T_{n,k} \Vdash_\varphi \mathbf{v}^i$, $i = 1 \ldots n$, *and* $SatLoc(\varphi, \mathbf{v}, \mathbf{v}^1, \ldots, \mathbf{v}^n) =$true.

(ii) $T^+_{n,k+1} \Vdash_\varphi (\mathbf{v}, \mathbf{u}) \Leftrightarrow$ *there exist* $\mathbf{v}^1, \ldots, \mathbf{v}^n \in \{0,1\}^n$ *such that* $T^+_{n,k} \Vdash_\varphi (\mathbf{v}^i, \mathbf{u})$, $i = 1 \ldots n$, *and* $SatLoc(\varphi, \mathbf{v}, \mathbf{v}^1, \ldots, \mathbf{v}^n) =$true.

Proof. By definition,

$$T_{n,k+1} = F + (G_1 \sqcup \ldots \sqcup G_n),\ T^+_{n,k+1} = T_{n,k+1} + F',$$

where $F, F' \in \mathcal{T}_{n,1}$, $G_1, \ldots, G_n \in \mathcal{T}_{n,k}$. Let $x, y_1, \ldots, y_n, z$ be the roots of $F, G_1, \ldots, G_n, F'$ respectively, $\mathbf{x} := (x, y_1, \ldots, y_n)$, $\mathbf{y} := (y_1, \ldots, y_n)$ (Figure 2a).

(i) ($\Rightarrow$). For some θ we have: $T_{n,k+1}, \theta, x \vDash \varphi^{\mathbf{v}}$. Thus $T_{n,k+1}, \theta, y_i \vDash \varphi^{\mathbf{v}^i}$ for some vectors $\mathbf{v}^1, \ldots, \mathbf{v}^n \in \{0,1\}^n$. Obviously, $T_{n,k} \Vdash_\varphi \mathbf{v}^i$ for all $i = 1 \ldots n$. Since $T_{n,k+1}, \mathbf{y} \Vdash_\varphi (\mathbf{v}^1, \ldots, \mathbf{v}^n)$ and $T_{n,k+1}, \mathbf{x} \Vdash_\varphi (\mathbf{v}, \mathbf{v}^1, \ldots, \mathbf{v}^n)$, we have $SatLoc(\varphi, \mathbf{v}, \mathbf{v}^1, \ldots, \mathbf{v}^n)$=true.
($\Leftarrow$). It is not difficult to see that $T_{n,k+1}, \mathbf{y} \Vdash_\varphi (\mathbf{v}^1, \ldots, \mathbf{v}^n)$.
Since $SatLoc(\varphi, \mathbf{v}, \mathbf{v}^1, \ldots, \mathbf{v}^n)$=true, we have $T_{n,k+1}, \mathbf{x} \Vdash_\varphi (\mathbf{v}, \mathbf{v}^1, \ldots, \mathbf{v}^n)$, so $T_{n,k+1} \Vdash_\varphi \mathbf{v}$.

(ii) ($\Rightarrow$). Similar to (i).
($\Leftarrow$). For some valuations $\theta_1, \ldots, \theta_n$ we have:

$$T^+_{n,k+1}, \theta_i, y_i \vDash \varphi^{\mathbf{v}^i},\ T^+_{n,k+1}, \theta_i, z \vDash \varphi^{\mathbf{u}}.$$

We define a valuation θ as follows:

$$\theta(p) := \bigcup_i \{y \in G_i \mid y \in \theta_i(p)\} \cup \{y \in F' \mid y \in \theta_1(p)\}.$$

A straightforward argument shows that $T_{n,k+1}, \theta, y_i \vDash \varphi^{\mathbf{v}^i}$ for all $i = 1 \ldots n$, so $T_{n,k+1}, \mathbf{y} \Vdash_\varphi (\mathbf{v}^1, \ldots, \mathbf{v}^n)$. $SatLoc(\varphi, \mathbf{v}, \mathbf{v}^1, \ldots, \mathbf{v}^n)$=$\mathsf{true}$ implies $T^+_{n,k+1}, \mathbf{x} \Vdash_\varphi (\mathbf{v}, \mathbf{v}^1, \ldots, \mathbf{v}^n)$, i.e. for some valuation η we have

$$T^+_{n,k+1}, \eta, x \vDash \varphi^{\mathbf{v}},\ T^+_{n,k+1}, \eta, y_i \vDash \varphi^{\mathbf{v}^i} \text{ for all } i = 1 \ldots n.$$

We put

$$\eta'(p) := \{y \in F \mid x \in \eta(p)\} \cup \{x \notin F \mid x \in \theta(p)\}.$$

One can check that $T^+_{n,k+1}, \eta', x \vDash \varphi^{\mathbf{v}}$ and $T^+_{n,k+1}, \eta', z \vDash \varphi^{\mathbf{u}}$, that is $T_{n,k+1} \Vdash_\varphi (\mathbf{v}, \mathbf{u})$. ∎

Now let us give a recursive description of the algorithms $SatTree$ and $SatTree^+$ determining whether $T_{n,k} \Vdash_\varphi \mathbf{v}$ and $T^+_{n,k} \Vdash_\varphi (\mathbf{v}, \mathbf{u})$ (for the basic case $k = 1$ these algorithms - SAT_1 and SAT^+_1 are constructed in Appendix).

Function $SatTree(\varphi, \mathbf{v}, k)$ returns boolean
Begin
  if $k = 1$ then return($SAT_1(\varphi, \mathbf{v})$);
  for all $\mathbf{v}^1, \ldots, \mathbf{v}^n \in \{0,1\}^n$:
    if $\bigwedge_{1 \le i \le n} SatTree(\varphi, \mathbf{v}^i, k-1) \bigwedge SatLoc(\varphi, \mathbf{v}, \mathbf{v}^1, \ldots, \mathbf{v}^n)$
      then return(true);
  return(false);
End.

Function $SatTree^+(\varphi, \mathbf{v}, \mathbf{u}, k)$ returns boolean
Begin
  if $k = 1$ then return($SAT^+_1(\varphi, \mathbf{v}, \mathbf{u})$);
  for all $\mathbf{v}^1, \ldots, \mathbf{v}^n \in \{0,1\}^n$:
    if $\bigwedge_{1 \le i \le n} SatTree^+(\varphi, \mathbf{v}^i, \mathbf{u}, k-1) \bigwedge SatLoc(\varphi, \mathbf{v}, \mathbf{v}^1, \ldots, \mathbf{v}^n)$
      then return(true);
  return(false);
End.

By Lemma 17, we obtain

LEMMA 18.

(i) $T_{n,k} \Vdash_{\varphi} \mathbf{v} \Leftrightarrow SatTree(\varphi, \mathbf{v}, k) = \mathsf{true}$.

(ii) $T^{+}_{n,k} \Vdash_{\varphi} (\mathbf{v}, \mathbf{u}) \Leftrightarrow SatTree^{+}(\varphi, \mathbf{v}, \mathbf{u}, k) = \mathsf{true}$.

Function $SAT\mathbf{L}_1(\varphi)$ returns boolean
Begin
  for all $\mathbf{v} \in \{0,1\}^n$, $\mathrm{v}_n = 1$:
    if $SatTree(\varphi, \mathbf{v}, n)$ then return(true);
  return(false);
End.

Function $SAT\mathbf{L}_2(\varphi)$ returns boolean
Begin
  for all $\mathbf{v}, \mathbf{u} \in \{0,1\}^n$, $\mathrm{v}_n = 1$:
    if $SatTree^{+}(\varphi, \mathbf{v}, \mathbf{u}, n)$ then return(true);
  return(false);
End.

By Lemma 16 and Lemma 18, we obtain

THEOREM 19. *φ is $\mathbf{L}_i$-satisfiable $\Leftrightarrow$ $SAT\mathbf{L}_i(\varphi)$=true, $i = 1, 2$.*

One can see that the space used on each level of recursion is $O(n^2)$. The depth of recursion is n, and the total amount of space required is $O(n^3)$.

Now let us show that the satisfiability problems for $\mathbf{L}_1$ and $\mathbf{L}_2$ are PSPACE-hard.

Since satisfiability problem for all logics between **K** and **S4** is PSPACE-hard (Ladner's Theorem [11]), we obtain that $\mathbf{L}_1$ is PSPACE-hard. Since $\mathbf{S4} \nvdash A2$, the logic $\mathbf{L}_2 \not\subseteq \mathbf{S4}$. However, the following slight modification of Ladner's construction [11] proves the PSPACE-hardness for all logics between **K4** and **S4.1.2**.

Let A be a propositional logic formula, $PV(A) = \{p_1, \ldots, p_n\}$, and $B = Q_1 p_1 \ldots Q_n p_n A$, where $Q_1, \ldots, Q_n \in \{\forall, \exists\}$. We put

$$\phi_1(B) := \bigwedge_{0 \le i \le n} (q_i \to \Diamond(\neg q_i \wedge q_{i+1})),$$

$$\phi_2(B) := \bigwedge_{\{i \mid Q_i = \forall\}} (q_{i-1} \to \Diamond(q_i \wedge p_i) \wedge \Diamond(q_i \wedge \neg p_i)),$$

$$\phi_3(B) := \bigwedge_{1 \le i \le n} ((q_i \wedge p_i \to \Box(q_n \to p_i)) \wedge (q_i \wedge \neg p_i \to \Box(q_n \to \neg p_i))),$$

$$\varphi(B) := q_0 \wedge \Box(q_n \to A) \wedge \Box(\phi_1(B) \wedge \phi_2(B) \wedge \phi_3(B)).$$

A straightforward argument shows that

B is valid $\Rightarrow \varphi(B)$ is **S4.1.2**-satisfiable;

$\varphi(B)$ is **K4**-satisfiable $\Rightarrow B$ is valid.

Since the validity problem for prenex quantified boolean formulas is PSPACE-complete [16], we obtain

THEOREM 20. *If* $\mathbf{K4} \subseteq \mathbf{L} \subseteq \mathbf{S4.1.2}$*, then the satisfiability problem for* $\mathbf{L}$ *is PSPACE-hard.*[4]

Note that $\mathbf{K4} \subset \mathbf{L}_2 \subset \mathbf{S4.1.2}$. So by Theorems 19, 20, we obtain

THEOREM 21. $\mathbf{L}_1$*,* $\mathbf{L}_2$ *are PSPACE-complete.*

7 Examples

In this section we illustrate our method with some examples. We consider the logics $\mathbf{K4}, \mathbf{K4d}, \mathbf{S4}$ and their extensions by confluence and McKinsey axiom. These logics are known to be in PSPACE[5] (cf. [11],[3],[9]). Our method yields an alternative proof of this fact.

Consider a logic $\Lambda = \mathbf{L}(\mathcal{F}^\Lambda)$. To prove the PSPACE-decidability of Λ it is sufficient to show that for any formula φ there exists a class $\mathcal{F}^\Lambda_\varphi \subseteq \mathcal{F}^\Lambda$ such that:

- φ is Λ-satisfiable $\Rightarrow \varphi$ is $\mathcal{F}^\Lambda_\varphi$-satisfiable;
- It is possible to decide whether φ is $\mathcal{F}^\Lambda_\varphi$-satisfiable within the space polynomial of $|Sub(\varphi)|$.

In many cases, it is possible to present $\mathcal{F}^\Lambda_\varphi$ as a finite class of quasitrees (or quasitrees with some additional clusters on the top) with appropriate restriction of height, branching and thickness. Actually, the main problem is how to restrict the height of frames in $\mathcal{F}^\Lambda_\varphi$.

Let us reformulate Lemma 3:

LEMMA 22. *Let* $\mathfrak{M}$ *be the canonical model of a logic* $\Lambda \supseteq \mathbf{K4}$*, and assume that a formula* $\Diamond\psi$ *is satisfied at some* $x \in \mathfrak{M}$*. Let*

$$Y := \{y \mid xRy \ \&\ \mathfrak{M}, y \models \Diamond\psi\} \cup \{x\}.$$

Then the model $\mathfrak{M}|Y$ *contains a maximal cluster.*

[4] This statement actually holds for all logics between $\mathbf{K}$ and $\mathbf{S4.1.2}$. The proof is by an easy modification of φ.

[5] Moreover, they are PSPACE-complete.

Using this lemma, it is not difficult to check that for every logic $\Lambda \supseteq \mathbf{K4}$ and formula φ it is possible to extract a selective filtration M from the canonical model of Λ through $Sub(\varphi)$ such that $\mathbf{h}(M) = O(|Sub(\varphi)|)$.

However, it is necessary to show that we obtain a Λ-frame. For example, consider $\Lambda \supseteq \mathbf{K4d}$. It is not difficult to see that in this case every maximal cluster in Y is non-degenerate. It allows us to obtain a dense selective filtration, which implies the result for $\mathbf{K4d}$. For the logics $\mathbf{K4d.1}$, $\mathbf{K4d.2}$, $\mathbf{K4d.1.2}$ we modify the construction as we did in Lemma 12 (ii) for the logic $\mathbf{L}_2$, i.e. we extract an additional final cluster (or clusters) from the canonical model.

Let $\mathcal{C}(n) := \{C \mid C \text{ is a non-degenerate cluster}, |C| \leq n\}$. For a frame G let G^{MK} denote the frame obtained by putting a reflexive singleton above each maximal cluster in G.

For $\Lambda = \mathbf{K4}, \mathbf{K4d}$ we put:

$$\mathcal{G}^{\Lambda}(n) := \{G \mid G \text{ is } \Lambda\text{-quasitree}, \mathbf{h}(G) \leq 2n,\ \mathbf{b}(G) \leq n,\ \mathbf{t}(G) \leq n\};$$
$$\mathcal{G}^{\Lambda.1}(n) := \{G^{MK} \mid G \in \mathcal{G}^{\Lambda}(n)\};$$
$$\mathcal{G}^{\Lambda.2}(n) := \{G + C \mid G \in \mathcal{G}^{\Lambda}(n),\ C \in \mathcal{C}(n)\} \cup \{C_0\};$$
$$\mathcal{G}^{\Lambda.1.2}(n) := \{G + C \mid G \in \mathcal{G}^{\Lambda}(n),\ C \text{ is a reflexive singleton}\};$$

Similar to Lemma 13, one can check:

LEMMA 23. *Consider the logic*
$\Lambda \in \{\mathbf{K4},\ \mathbf{K4.1},\ \mathbf{K4.2},\ \mathbf{K4.1.2},\ \mathbf{K4d},\ \mathbf{K4d.1},\ \mathbf{K4d.2},\ \mathbf{K4d.1.2}\}$.
For a formula φ such that $|Sub(\varphi)| = n$ we have:
φ is Λ-satisfiable $\Leftrightarrow$ φ is $\mathcal{G}^{\Lambda}(n)$-satisfiable.

Note that this lemma is sufficient for establishing the PSPACE-decidability of these logics, so by Theorem 20 they are PSPACE-complete.

Sometimes, to check the satisfiability, one can use a single frame (as it was in the case of $\mathbf{L}_1$, $\mathbf{L}_2$). For example, consider the logic $\mathbf{S4}$. It is easy to modify the construction in Lemma 12 (i) to obtain an appropriate model for $\mathbf{S4}$: in the case of 2-dense logics degenerate clusters arise, and in the reflexive case these clusters are reflexive singletons. To obtain the frame with McKinsey property (or confluence, or both), we proceed as in Lemma 12 (ii). Similar to Lemmas 13,14, we obtain the following construction.

Let $\mathcal{T}^{S4}_{n,1}$ be the class of all frames isomorphic to $C_1 + C_n$. We put

$$\mathcal{T}^{S4}_{n,k+1} := \{F + (G_1 \sqcup \ldots \sqcup G_n) \mid F \in \mathcal{T}^{S4}_{n,1},\ G_1, \ldots, G_n \in \mathcal{T}^{S4}_{n,k}\},$$

$$\mathcal{T}^{S4.1}_{n,k} := \{F^{Mc} \mid F \in \mathcal{T}^{S4}_{n,k}\},$$

$$\mathcal{T}^{S4.2}_{n,k} := \{F + F' \mid F \in \mathcal{T}^{S4}_{n,k},\ F' \in \mathcal{T}^{S4}_{n,1}\},$$

$$\mathcal{T}^{S4.1.2}_{n,k} := \{F + C \mid F \in \mathcal{T}^{S4}_{n,k},\ C \text{ is a reflexive singleton}\}.$$

Let $T^{\Lambda}_{n,k} \in \mathbf{T}^{\Lambda}_{n,k}$.

LEMMA 24. *Consider the logic* $\Lambda \in \{\mathbf{S4},\ \mathbf{S4.1},\ \mathbf{S4.2},\ \mathbf{S4.1.2}\}$.
For a formula φ *such that* $|Sub(\varphi)| = n$ *we have:*
φ *is* Λ*-satisfiable* $\Leftrightarrow$ φ *is satisfiable at the root of* $T^{\Lambda}_{n,n}$.

A slight modification of the algorithms $SatLoc$, $SatTree$, $SatTree^+$ allows us to check the Λ-satisfiability in $O(|Sub(\varphi)|^3)$ amount of space, so we obtain that $\mathbf{S4}, \mathbf{S4.1}, \mathbf{S4.2}, \mathbf{S4.1.2}$ are PSPACE-complete.

In this paper we consider only transitive logics. However, sometimes our method can be used in the non-transitive case. For example, consider the logic of weak transitivity

$$\mathbf{K4^0} := \mathbf{K} + \Diamond\Diamond p \to \Diamond p \vee p,$$

axiomatizing derivation in arbitrary topological spaces [5]. The canonical model of $\mathbf{K4^0}$ has the maximality property, similarly to the transitive canonical model. This allows us to obtain a finite weakly transitive selective filtration of $\mathbf{K4^0}$ satisfying a given formula. Moreover, basing on the ideas of this paper, we showed that $\mathbf{K4^0}$ is in PSPACE, the proof will be published in the sequel.

8 Appendix

Function $SatLoc(\varphi, \mathbf{v}, \mathbf{v}^1, \ldots, \mathbf{v}^n)$ returns boolean
Begin
`REM`{ *For* $\theta \in \{0,1\}^{(n+1)\times m}$ *we construct* $\eta \in \{0,1\}^{(n+1)\times n}$, where
θ^i_j *is the trues value of* p_j *at* x_i*,*
η^i_j *is the trues value of* ψ_j *at* x_i*, Figure 2b.*}

```
for all θ ∈ {0,1}^((n+1)×m):
begin
  for j := 1...n:                                REM{ ψ_j ∈ Sub(φ)}
  begin
    for i := 0...n:                              REM{ x_i ∈ C_0 + C_n}
    begin
      η^i_j := 0;
      if j ≤ m then η^i_j := θ^i_j;              REM{ ψ_j is a variable}
      if ψ_j = ψ_s → ψ_l then                    REM{ note that s,l < j}
        if η^i_s = 0 or η^i_l = 1 then η^i_j := 1;
      if ψ_j = ◊ψ_s then                         REM{ note that s < j}
```

begin
for $l := 1 \ldots n$: if $\eta_s^l = 1$ then $\eta_j^i := 1$;
for $l := 1 \ldots n$: if $\mathrm{v}_s^l = 1$ or $\mathrm{v}_j^l = 1$ then $\eta_j^i := 1$;
end;
end;
end;
if $(\eta_1^0, \ldots, \eta_n^0) = \mathbf{v}$ then return(true);
end;
return(false);
End.

Function $SAT_1(\varphi, \mathbf{v})$ returns boolean
Begin return($SatLoc(\varphi, \mathbf{v}, \bar{0}, \ldots, \bar{0})$); End.

Function $SAT_1^+(\varphi, \mathbf{v}, \mathbf{u})$ returns boolean
Begin return($SatLoc(\varphi, \mathbf{v}, \mathbf{u}, \bar{0}, \ldots, \bar{0}) \bigwedge SAT_1(\varphi, \mathbf{u})$); End.

9 Acknowledgements

The author is grateful to prof. Valentin Shehtman for his help and also to the anonymous referees for their useful comments.

The work on this paper was supported by Poncelet Laboratory (UMI 2615 of CNRS and Independent University of Moscow), RFBR (project No.02-01-22003), and by CNRS (ECO-NET 2004, project No. 08111TL).

BIBLIOGRAPHY

[1] E. W. Beth. The foundation of mathemathics. North-Holland, Amsterdam, 1959.
[2] P. Blackburn, M. de Rijke and Y. Venema. Modal logic. Cambridge University Press, 2001
[3] L. Farinas del Cerro and O. Gasquet. A general framework for pattern-driven modal tableaux. Logic Journal of the IGPL, 10(1):51-83, 2002.
[4] A. Chagrov, M. Zakharyaschev. Modal logic. Oxford University Press, 1997.
[5] L. Esakia. Weak transitivity - a restitution (in Russian). In: Logical investigations, v.8, p.244-245. Moscow, Nauka, 2001.
[6] K. Fine. Logics containing K4, part II. Journal of Symbolic Logic, 50: 619-651, 1985.
[7] R. Goldblatt. Diodorean modality in Minkowski spacetime. Studia Logica, v. 39 (1980), 219-236.
[8] J. Halpern and Y.Moses. A guide to completeness and complexity for modal logics of knowledge and belief. Artificial Intelligence, 54:319-379, 1992.
[9] Marcus Kracht. Notes on the space requirements for checking satisfiability in modal Logics. Advances in Modal Logic, Volume 4, 243-264. King's College Publications, 2003.
[10] S. Kripke. A semantical analysis of modal logic I: Normal modal propositional calculi. Z. Math. Logik Grundl. Math., 9:67-96, 1963.
[11] R. Ladner. The computational complexity of provability in systems of modal propositional logic. SIAM Journal of Computing, 6:467-480, 1977.

[12] I. Shapirovsky, V. Shehtman. Chronological future modality in Minkowski spacetime. Advances in Modal Logic, Volume 4, 437-459. King's College Publications, 2003.
[13] V.B. Shehtman. Modal logics of domains on the real plane. Studia Logica, v. 42 (1983), 63-80.
[14] E. Spaan. Complexity of modal logics. PhD thesis, University of Amsterdam, 1993.
[15] E. Spaan. The complexity of propositional tense logics. In de Rijke, M. (Ed.), Diamonds and Defaults, pp. 287-307. Kluwer Academic Publishers, Dordrecht, 1993.
[16] L. J. Stockmeyer and A. R. Meyer. Word problems requiring exponential time: preliminary report. In Proc. 5th ACM Symp. on Theory of Computing, pp.1-9, 1973.

Ilya Shapirovsky

Institute for Information Transmission Problems
Russian Academy of Sciences,
B.Karetny 19, Moscow, Russia, 101447

E-mail: ilshapir@netscape.net