Moving in a Crumbling Network: The Balanced Case

Philipp Rohde

RWTH Aachen, Informatik VII
rohde@informatik.rwth-aachen.de

Abstract. In this paper we continue the study of 'sabotage modal logic' SML which was suggested by van Benthem. In this logic one describes the progression along edges of a transition graph in alternation with moves of a saboteur who can delete edges. A drawback of the known results on SML is the asymmetry of the two modalities of 'moving' and 'deleting': Movements are local, whereas there is a global choice for edge deletion. To balance the situation and to obtain a more realistic model for traffic and network problems, we require that also the sabotage moves (edge deletions) are subject to a locality condition. We show that the new logic, called path sabotage logic PSL, already has the same complexities as SML (model checking, satisfiability) and that it lacks the finite model property. The main effort is finding a pruned form of SML-models that can be enforced within PSL and giving appropriate reductions from SML to PSL.

Keywords: modal logics, dynamic logics, model checking

1 Introduction

In the 'classical' framework of model checking one considers movements of agents within a system, but the underlying structure is assumed to be static. So in many formalisms only properties of unchanged systems are expressible. This motivates a more general approach where dynamic changes of the underlying structure are relevant. For example, consider a computer network where connections may break down. Some natural questions arise for such a system: Is it possible – regardless of the removed connections – to interchange information between two designated servers? Another task of this kind arises for navigation systems: Is it possible to find a way between cities within a traffic network where connections are canceled, e.g., because of roadworks or traffic jams?

To specify problems of this nature, van Benthem considered 'sabotage modal logics' which are modal logics over changing models (cf. [1]). He introduced a cross-model modality referring to submodels from which objects have been removed. SML consists of standard modal logic equipped with a 'edge-deleting' modality and is capable of expressing elementary changes of transition systems itself. One could express problems related to this situation by first order specifications, but then one has to put up with the high complexity of FO. So SML seems to be a moderate strengthening of modal logic for this kind of problems.

But in [3] and [4] we showed that the new operator already strengthens modal logic in such a way that all the nice algorithmic and model-theoretic properties of modal logic get lost. In fact, from the viewpoint of complexity, SML much more resembles FO than modal logic: Uniform model checking for SML is PSPACE-complete and the satisfiability problem is undecidable. But after all, an advantage of SML over FO is a linear formula and a polynomial program complexity of model checking.

A drawback of SML is the asymmetry of the two modalities of 'moving' and 'deleting': Movements are local, whereas the choice for edge deletion is global. So SML seems to be an appropriate specification for dynamic problems like the traffic problem mentioned above: The canceling of connections is global and (almost) independent of a movement within the system. But for other dynamic tasks SML fails to be a realistic model, especially if the 'saboteur' also has to move within the system using the same connections as the 'runner'. For example, a computer virus needs to use the same internet connections before it reaches the target that it wants to block. In this paper we introduce the path sabotage logic PSL to balance the situation: We require that the saboteur moves within the system such that exactly those edges are deleted that were taken along his path. Hence also the sabotage moves are subject to a locality condition. We show that PSL already has the same complexities as SML and that PSL also fails to have the finite model property.

In Sect. 2 we repeat the definition of SML and introduce the logic PSL. In Sect. 3 we show that model checking for PSL is PSPACE-complete and that PSL has an effective formula and program complexity. To reduce the satisfiability problem for SML to the same problem for PSL we need a kind of normal form for SML-models (relative to a given SML-formula), namely pruned models. In Sect. 4 we introduce this notion and show that every SML-model can be transformed into a pruned form. In Sect. 5 we show how to enforce within PSL that a model of a given SML-formula contains a pruned submodel together with some additional properties that we need for the reduction of the satisfiability problem.

I would like to thank Christof Löding for several comments and Benedikt Löwe who had the idea of the path sabotage logic.

2 Preliminaries

In this section we repeat the definition of the sabotage modal logic SML with a global 'edge-deleting' modality and introduce the balanced version of SML with a 'deleting by moving' modality which we call path sabotage logic PSL. We interpret both logics over edge-labeled transition systems. For that let Prop be a finite set of unary predicate symbols. A transition system $\mathcal{T}$ is a tuple (S, Σ, R, L) with a set of states S, a finite alphabet Σ, a ternary transition relation $R \subseteq S \times \Sigma \times S$ and a labeling function $L : S \to 2^{\mathrm{Prop}}$.

Let $p \in \mathrm{Prop}$ and $a \in \Sigma$. Formulae of the *sabotage modal logic* SML are inductively defined by the grammar

$$\varphi ::= \top \mid p \mid \neg\varphi \mid \varphi \vee \varphi \mid \Diamond_a \varphi \mid \not\Diamond_a \varphi.$$

As usual, $\bot$ is an abbreviation for $\neg\top$. The dual modalities are defined by $\Box_a\varphi := \neg\Diamond_a\neg\varphi$ and $\boxminus_a\varphi := \neg\diamondminus_a\neg\varphi$.

Let $\mathcal{T} = (S, \Sigma, R, L)$ be a transition system. For a set $E \subseteq R$ we define the transition system $\mathcal{T} \setminus E := (S, \Sigma, R \setminus E, L)$. The semantics of SML relative to a current position $s \in S$ are inductively defined by

$$
\begin{array}{ll}
(\mathcal{T}, s) \models \top & \text{is true,} \\
(\mathcal{T}, s) \models p & \text{iff } p \in L(s), \\
(\mathcal{T}, s) \models \neg\varphi & \text{iff not } (\mathcal{T}, s) \models \varphi, \\
(\mathcal{T}, s) \models \varphi \vee \psi & \text{iff } (\mathcal{T}, s) \models \varphi \text{ or } (\mathcal{T}, s) \models \psi, \\
(\mathcal{T}, s) \models \Diamond_a\varphi & \text{iff there is } s' \in S \text{ with } (s, a, s') \in R \text{ and } (\mathcal{T}, s') \models \varphi, \\
(\mathcal{T}, s) \models \diamondminus_a\varphi & \text{iff there is } (t, a, t') \in R \text{ with } (\mathcal{T} \setminus \{(t, a, t')\}, s) \models \varphi.
\end{array}
$$

The sabotage modality $\diamondminus$ has the global power to delete transitions somewhere in the system whereas the standard modality $\Diamond$ only allows of moving locally. To balance the situation we introduce a new sabotage modality $\diamonddot$ such that deletion is combined with a movement that is independent of the one according to the standard modalities. Hence a current position in the system becomes a pair of states. The syntax of the *path sabotage logic* PSL is defined in the same way, but using the modality $\diamonddot_a$ instead of $\diamondminus_a$, for $a \in \Sigma$. The dual modality $\boxdot_a$ is defined analogously.

The semantics of PSL relative to a current position $[s, t]$ for $s, t \in S$ are inductively defined by

$$
\begin{array}{ll}
(\mathcal{T}, s, t) \models \top & \text{is true,} \\
(\mathcal{T}, s, t) \models p & \text{iff } p \in L(s), \\
(\mathcal{T}, s, t) \models \neg\varphi & \text{iff not } (\mathcal{T}, s, t) \models \varphi, \\
(\mathcal{T}, s, t) \models \varphi \vee \psi & \text{iff } (\mathcal{T}, s, t) \models \varphi \text{ or } (\mathcal{T}, s, t) \models \psi, \\
(\mathcal{T}, s, t) \models \Diamond_a\varphi & \text{iff there is } s' \in S \text{ with } (s, a, s') \in R \text{ and } (\mathcal{T}, s', t) \models \varphi, \\
(\mathcal{T}, s, t) \models \diamonddot_a\varphi & \text{iff there is } t' \in S \text{ with } (t, a, t') \in R \text{ and} \\
& (\mathcal{T} \setminus \{(t, a, t')\}, s, t') \models \varphi.
\end{array}
$$

Note that propositions can only be checked on paths built up by standard modalities.

A measure for the complexity of an SML-formula φ is the number of nested sabotage modalities. We call this the *sabotage depth* $\mathrm{sd}(\varphi)$ of φ and define inductively

$$
\begin{array}{rr}
\mathrm{sd}(\top) := \mathrm{sd}(p) := 0, & \mathrm{sd}(\varphi_1 \vee \varphi_2) := \max\{\mathrm{sd}(\varphi_1), \mathrm{sd}(\varphi_2)\}, \\
\mathrm{sd}(\neg\psi) := \mathrm{sd}(\Diamond_a\psi) := \mathrm{sd}(\psi), & \mathrm{sd}(\diamondminus_a\psi) := \mathrm{sd}(\psi) + 1.
\end{array}
$$

The number of nested path sabotage operators of a PSL-formula φ is called *path sabotage depth* $\mathrm{pd}(\varphi)$ and is defined analogously.

For a fixed $a \in \Sigma$, the number $\mathrm{sd}_a(\varphi)$ of nested modalities $\diamondminus_a$ is defined in the same way, but using $\mathrm{sd}_a(\diamondminus_a\psi) := \mathrm{sd}_a(\psi) + 1$ and $\mathrm{sd}_a(\diamondminus_b\psi) := \mathrm{sd}_a(\psi)$ for

$b \neq a$. In the next section we will see that the path sabotage depth $\mathrm{pd}(\varphi)$ of a formula φ is the main factor in the complexity of the model checking problem for PSL. But first we repeat some known results on the logic SML. The combined complexity of SML model checking, i.e., the complexity measured in terms of the size of the formula and the size of the structure, was already settled in [3]. The formula and program complexity of SML model checking was determined in [4].

Theorem 1. *1. Combined complexity: Model checking for* SML *is* PSPACE-*complete.*
2. *Formula complexity: Model checking for* SML *with a fixed transition system can be solved in linear time in the size of the formula.*
3. *Program complexity: Model checking for a fixed* SML-*formula can be solved in polynomial time in the size of the transition system.* □

In [4] it was also shown that, in contrast to modal logic where each satisfiable formula has a finite model, this property does not hold for SML.

Theorem 2. *There is an* SML-*formula that has only infinite models.* □

Further it was proven that the satisfiability problem for SML is undecidable. To be more precise:

Theorem 3. *The problems of deciding whether a given* SML-*formula has a model (Satisfiability), has a finite model (Finite Satisfiability), or is satisfiable, but only has infinite models (Infinity Axiom) are undecidable.* □

3 Model Checking for PSL

In this section we show that model checking for PSL is also PSPACE-complete. For membership we give a translation of PSL into first order logic. The completeness is shown by a reduction of the SML model checking problem to the one for PSL. In the rest of the section we show that PSL has an effective formula and program complexity. We do that by translating the model checking problem for PSL into the one for standard modal logic. Some proofs are slight modifications of the ones for SML that are presented in [3] and [4], so we omit the details.

By heavy use of variables one can translate PSL into first order logic. Since FO model checking is in PSPACE we obtain:

Lemma 4. *For every* PSL-*formula* φ *there is an effectively constructible* FO-*formula* $\hat{\varphi}(x, y)$ *such that for every transition system* $\mathcal{T}$ *and states* s, t *of* $\mathcal{T}$ *one has:*

$$(\mathcal{T}, s, t) \models \varphi \iff \mathcal{T} \models \hat{\varphi}[s, t].$$

The size of $\hat{\varphi}(x, y)$ *is polynomial in the size of* φ. *In particular, model checking for* PSL *is in* PSPACE. □

Next we give a reduction of SML model checking to PSL model checking. For an alphabet Σ and $m \geq 1$ let $\Sigma_m := \Sigma \,\dot{\cup}\, \{1, \ldots, m\}$ (w.l.o.g. we assume that $i \notin \Sigma$ for every $1 \leq i \leq m$). For a transition system $\mathcal{T} = (S, \Sigma, R, L)$ we define the transition system $\mathcal{T}_m := (S, \Sigma_m, R_m, L)$, where

$$R_m := R \,\dot{\cup}\, \{(s, i, s') \mid s, s' \in S \wedge 1 \leq i \leq m\}.$$

For $m = 0$ let $\Sigma_0 = \Sigma$ and $\mathcal{T}_0 = \mathcal{T}$. For a given SML-formula φ over Σ let the PSL-formula $\varphi^{\#}$ over $\Sigma_{\mathrm{sd}(\varphi)}$ be inductively defined as follows: $(\top)^{\#} := \top$, $(p)^{\#} := p$ and the operator $\#$ is homeomorphic for $\vee, \neg$ and $\Diamond_a$. For $\varphi = \diamondminus_a \psi$ and $i = \mathrm{sd}(\varphi)$ let $\varphi^{\#} := \diamonddot_i \diamonddot_a \psi^{\#}$. Note that $|R_m| = |R| + m \cdot |S|^2$ and that $|\varphi^{\#}|$ is polynomial in $|\varphi|$.

Lemma 5. *For every* SML*-formula* φ*, transition system* $\mathcal{T}$*, and* $s, t \in \mathcal{T}$ *it holds*

$$(\mathcal{T}, s) \models \varphi \iff (\mathcal{T}_{\mathrm{sd}(\varphi)}, s, t) \models \varphi^{\#}.$$

Proof. By induction on the structure of φ. Let $m := \mathrm{sd}(\varphi)$. Since the standard modalities in φ do not speak about the symbols $1, \ldots, m$, the only interesting case is for $\varphi = \diamondminus_a \psi$. Let $\mathcal{T} = (S, \Sigma, R, L)$ with $(\mathcal{T}, s) \models \varphi$. Then there is $(u, a, u') \in R$ such that $(\mathcal{T} \setminus \{(u, a, u')\}, s) \models \psi$. Since $\mathrm{sd}(\psi) = m - 1$, it holds $((\mathcal{T} \setminus \{(u, a, u')\})_{m-1}, s, u') \models \psi^{\#}$ by induction. Clearly we have

$$\mathcal{T}_n \setminus \{(v, a, v')\} = (\mathcal{T} \setminus \{(v, a, v')\})_n \tag{1}$$

for any transition $(v, a, v') \in R$ and $n \in \mathbb{N}$. Hence $(\mathcal{T}_{m-1} \setminus \{(u, a, u')\}, s, u') \models \psi^{\#}$ and therefore $(\mathcal{T}_{m-1}, s, u) \models \diamonddot_a \psi^{\#}$. Since the symbol m does not occur in $\psi^{\#}$, we can arbitrarily add m-transitions to the model without affecting the truth of $\psi^{\#}$. So we also have $(\mathcal{T}_m \setminus \{(t, m, u)\}, s, u) \models \diamonddot_a \psi^{\#}$. Since (t, m, u) is a transition in $\mathcal{T}_m$ we get $(\mathcal{T}_m, s, t) \models \diamonddot_m \diamonddot_a \psi^{\#}$.

For the converse let $\varphi^{\#} = \diamonddot_m \diamonddot_a \psi^{\#}$ with $(\mathcal{T}_m, s, t) \models \varphi^{\#}$. Then there are $u, u' \in S$ with $(u, a, u') \in R$ such that

$$(\mathcal{T}_m \setminus \{(t, m, u), (u, a, u')\}, s, u') \models \psi^{\#}.$$

Since the symbol m does not occur in $\psi^{\#}$ and by (1) it holds

$$((\mathcal{T} \setminus \{(u, a, u')\})_{m-1}, s, u') \models \psi^{\#}.$$

By induction we have $(\mathcal{T} \setminus \{(u, a, u')\}, s) \models \psi$, hence $(\mathcal{T}, s) \models \diamondminus_a \psi$. $\square$

Corollary 6. *The model checking problem for* PSL *is* PSPACE*-complete.*

Proof. By Lemma 4, PSL model checking is in PSPACE. As noted above the size of $\varphi^{\#}$ is polynomial in $|\varphi|$ and the size of $\mathcal{T}_{\mathrm{sd}(\varphi)}$ is polynomial in $|\mathcal{T}|$ and $|\varphi|$. By the previous lemma we have a polynomial time reduction of the PSPACE-hard SML model checking to PSL model checking. $\square$

In the rest of the section we give a reduction of PSL model checking to the one for standard modal logic. For a transition system $\mathcal{T} = (S, \Sigma, R, L)$ we define the transition system $\mathcal{T}^\diamond := (S^\diamond, \Sigma^\diamond, R^\diamond, L^\diamond)$ that encodes all possible ways of sabotaging $\mathcal{T}$:

$$\begin{aligned} S^\diamond :=& S \times S \times 2^R, \quad \Sigma^\diamond := \Sigma \,\dot\cup\, \{\bar{a} \mid a \in \Sigma\},\\ R^\diamond :=& \{((s,t,E), a, (s',t,E)) \mid (s,a,s') \in R \setminus E\} \cup \\ & \{((s,t,E), \bar{a}, (s,t',E')) \mid (t,a,t') \in R \setminus E \wedge E' = E \cup \{(t,a,t')\}\},\\ L^\diamond(s,t,E) :=& L(s) \text{ for each } s,t \in S \text{ and } E \subseteq R. \end{aligned}$$

Over this system one can simulate the sabotage operator $\Diamond_a$ by using an $\bar{a}$-transition, i.e., by the modal operator $\Diamond_{\bar{a}}$. This motivates the following inductive definition of the ML-formula $\varphi^\diamond$ for a given PSL-formula φ: $(\top)^\diamond := \top$, $(p)^\diamond := p$ and the operator $\diamond$ is homeomorphic for $\vee, \neg$ and $\Diamond_a$. For $\varphi = \Diamond_a \psi$ let $\varphi^\diamond := \Diamond_{\bar{a}} \psi^\diamond$.

Recall that $\mathrm{pd}(\varphi)$ denotes the depth of nested path sabotage operators of a PSL-formula φ (cf. Sect. 2). If $\mathrm{pd}(\varphi)$ is small then we do not need the complete transition system $\mathcal{T}^\diamond$ to evaluate $\varphi^\diamond$. So, for $n \in \mathbb{N}$, we define $\mathcal{T}^\diamond_n$ to be the transition system $\mathcal{T}^\diamond$ restricted to the states (s,t,E) with $|E| \leq n$. Note that $\mathcal{T}^\diamond_n = \mathcal{T}^\diamond$ for $n \geq |R|$. The proof of the following lemma is a slight modification of the one for SML presented in [4].

Lemma 7. *For every* PSL*-formula φ, transition system $\mathcal{T}$, and $s, t \in \mathcal{T}$ it holds*

$$(\mathcal{T}, s, t) \models \varphi \iff (\mathcal{T}^\diamond_{\mathrm{pd}(\varphi)}, (s,t,\emptyset)) \models \varphi^\diamond. \quad \square$$

This reduction can be used to determine the formula complexity and the program complexity of PSL model checking:

Corollary 8.
1. *Formula complexity: Model checking for* PSL *with a fixed transition system can be solved in linear time in the size of the formula.*
2. *Program complexity: Model checking for* PSL *with a fixed formula can be solved in polynomial time in the size of the transition system.*

Proof. It is well known that the model checking problem for modal logic over transition systems can be solved in time $\mathcal{O}(|\psi| \cdot |\mathcal{T}|)$, where $|\psi|$ is the size of the given ML-formula ψ and $|\mathcal{T}|$ is the size of the given transition system $\mathcal{T}$ (cf. [2]). Hence, by Lemma 7, we can solve the model checking problem for a PSL-formula φ and $\mathcal{T}$ in time $\mathcal{O}(|\varphi^\diamond| \cdot |\mathcal{T}^\diamond_{\mathrm{pd}(\varphi)}|)$. From the definition of $\varphi^\diamond$ we get $|\varphi^\diamond| = |\varphi|$.

1. For a fixed transition system $\mathcal{T}$ we can estimate the size of $\mathcal{T}^\diamond_{\mathrm{pd}(\varphi)}$ by $|\mathcal{T}^\diamond_{\mathrm{pd}(\varphi)}| \in \mathcal{O}(|\mathcal{T}|^2 \cdot 2^{|\mathcal{T}|})$. Hence the formula complexity is in $\mathcal{O}(|\varphi|)$.

2. Since the number of subsets $E \subseteq R$ with $|E| \leq \mathrm{pd}(\varphi)$ is in $\mathcal{O}(|\mathcal{T}|^{\mathrm{pd}(\varphi)})$ we get $|\mathcal{T}^\diamond_{\mathrm{pd}(\varphi)}| \in \mathcal{O}(|\mathcal{T}|^{\mathrm{pd}(\varphi)+2})$. So the model checking complexity with a fixed PSL-formula φ is polynomial in $|\mathcal{T}|$. $\square$

4 Pruned SML-Models

In the last section we gave a reduction of the model checking problem for SML to the one for PSL. For a reduction of the satisfiability problem we need a more sophisticated approach. In this section we show that each model of a given SML-formula φ can be pruned such that it consists only of those states that are reachable from the initial state by the standard modalities in φ together with a bounded number of additional states (we call it a *pruned model relative to* φ). We define the pruned form of a model in two steps. In the next section we show how to enforce within PSL that a model of a given SML-formula φ contains a pruned submodel (relative to φ) where each two states are connected by i-transitions for $1 \leq i \leq \mathrm{sd}(\varphi)$ and such that one cannot escape the pruned submodel by using the modalities of φ. Then we can use the same argument as before to translate SML-modalities into PSL-modalities.

Let φ be an SML-formula over Σ. We define inductively the set of path labels $P_\varphi \subseteq \Sigma^*$ corresponding to the standard modalities in φ:

$$P_\varphi := \begin{cases} \{\varepsilon\} & \text{if } \varphi = \top \text{ or } \varphi = p, \\ P_{\varphi_1} \cup P_{\varphi_2} & \text{if } \varphi = \varphi_1 \vee \varphi_2, \\ P_\psi & \text{if } \varphi = \neg\psi \text{ or } \varphi = \diamondminus_a \psi, \\ \{\varepsilon\} \cup \{a \cdot \pi \mid \pi \in P_\psi\} & \text{if } \varphi = \Diamond_a \psi. \end{cases}$$

For $\mathcal{T} = (S, \Sigma, R, L)$ and $s \in \mathcal{T}$ let $\mathcal{T}_{\varphi,s} := (S_{\varphi,s}, \Sigma, R_{\varphi,s}, L|_{S_{\varphi,s}})$ be the transition system restricted to paths in P_φ starting in s:

$$\begin{aligned} S_{\varphi,s} &:= \{t \mid t \in S \text{ and there is a } \pi\text{-path from } s \text{ to } t \text{ in } \mathcal{T} \text{ for some } \pi \in P_\varphi\}, \\ R_{\varphi,s} &:= \{(t, a, t') \mid (t, a, t') \in R \text{ and there is a } \pi\text{-path from } s \text{ to } t \text{ in } \mathcal{T} \\ &\qquad \text{for some } \pi \in P_\varphi, \text{ such that } \pi \cdot a \in P_\varphi\}. \end{aligned}$$

Note that, if $(\mathcal{T}, s) \models \varphi$, then $\mathcal{T}_{\varphi,s}$ does not need to be a model of φ. There may be 'dummy' transitions in $\mathcal{T}$ that have to be deleted to satisfy φ, but which are not reachable by the standard modalities of φ.

Example 9. Consider the formula $\varphi := \Diamond_a \top \wedge \diamondminus_a \Box_a \bot \wedge \diamondminus_a \diamondminus_a \top$. The following transition system $(\mathcal{T}, s)$ is a model of φ:

$$s \xrightarrow{a} s' \xrightarrow{a} s''$$

Since $P_\varphi = \{\varepsilon, a\}$ the transition system $\mathcal{T}_{\varphi,s}$ consists only of the states s, s' and the transition (s, a, s'). Since we cannot delete two different a-transitions, it fails to be a model of φ.

But in fact, the exact position of a 'dummy' transition in $\mathcal{T}$ is irrelevant, hence we can equip $\mathcal{T}_{\varphi,s}$ with these transitions in a canonical way. Further we can bound the number of these transitions: One only needs $\mathrm{sd}_a(\varphi)$ many additional a-transitions for each $a \in \Sigma$, where $\mathrm{sd}_a(\varphi)$ is the depth of nested $\diamondminus_a$ in φ (cf. Sect. 2). We show this in the rest of the section.

For two sets R and R' of transitions with $R' \subseteq R$ and $a \in \Sigma$ let $\mathrm{diff}_a(R, R') := |R \setminus R' \cap S \times \{a\} \times S| \in \mathbb{N} \cup \{\infty\}$. Let $\kappa^a_{\varphi,s} \in \mathbb{N}$ be the minimum of $\mathrm{sd}_a(\varphi)$ and the number of a-transitions in $\mathcal{T}$ that are not present in $\mathcal{T}_{\varphi,s}$:

$$\kappa^a_{\varphi,s} := \min\{\mathrm{sd}_a(\varphi), \mathrm{diff}_a(R, R_{\varphi,s})\}.$$

The *pruned form* $\mathcal{T}^*_{\varphi,s}$ of an SML-model $\mathcal{T}$ relative to φ and s is defined by $\mathcal{T}^*_{\varphi,s} := (S^*_{\varphi,s}, \Sigma, R^*_{\varphi,s}, L_{\varphi,s})$, where

$$\begin{aligned} S^*_{\varphi,s} &:= S_{\varphi,s} \mathbin{\dot\cup} \{s^a_i \mid a \in \Sigma \land 1 \le i \le \kappa^a_{\varphi,s}\}, \\ R^*_{\varphi,s} &:= R_{\varphi,s} \mathbin{\dot\cup} \{(s^a_i, a, s) \mid a \in \Sigma \land 1 \le i \le \kappa^a_{\varphi,s}\}. \end{aligned}$$

Example 10. For the formula φ and the model $\mathcal{T}$ of Example 9 the transition system $\mathcal{T}^*_{\varphi,s}$ is

$$s^a_1 \xrightarrow{a} s \xrightarrow{a} s'$$

Theorem 11. *For every* SML*-formula φ, transition system $\mathcal{T}$, and $s \in \mathcal{T}$ it holds*

$$(\mathcal{T}, s) \models \varphi \iff (\mathcal{T}^*_{\varphi,s}, s) \models \varphi.$$

Proof. By induction on the structure of φ. For the atomic cases $\varphi = \top$ and $\varphi = p$ we have $P_\varphi = \{\varepsilon\}$ and $\kappa^a_{\varphi,s} = 0$ for every $a \in \Sigma$. Hence $S^*_{\varphi,s} = \{s\}$ and $(\mathcal{T}, s)$ is a model of φ iff $(\mathcal{T}^*_{\varphi,s}, s)$ is a model of φ. By induction and the fact that $\mathcal{T}^*_{\psi,s} \cong \mathcal{T}^*_{\neg\psi,s}$ the case $\varphi = \neg\psi$ is also clear.

Claim 1. For $\varphi = \psi \lor \chi$ it holds $(\mathcal{T}^*_{\varphi,s})^*_{\psi,s} \cong \mathcal{T}^*_{\psi,s}$ and $(\mathcal{T}^*_{\varphi,s})^*_{\chi,s} \cong \mathcal{T}^*_{\chi,s}$.

Proof (of Claim). By symmetry it is enough to show the first statement. Since $P_\psi \subseteq P_\varphi$ it is easy to see that $S_{\psi,s} \subseteq S_{\varphi,s}$ and $R_{\psi,s} \subseteq R_{\varphi,s}$. Since the additional states s^a_i in $\mathcal{T}^*_{\varphi,s}$ do not belong to $S_{\varphi,s}$ it follows $(\mathcal{T}^*_{\varphi,s})_{\psi,s} \cong \mathcal{T}_{\psi,s}$. Hence it suffices to show that the same number of additional states s^a_i is added to both models, for each $a \in \Sigma$. For that let $a \in \Sigma$ and λ^a be the number of states s^a_i in $(\mathcal{T}^*_{\varphi,s})^*_{\psi,s}$:

$$\lambda^a := \min\{\mathrm{sd}_a(\psi), \mathrm{diff}_a(R^*_{\varphi,s}, R_{\psi,s})\}.$$

Case 1: $\mathrm{sd}_a(\varphi) \le \mathrm{diff}_a(R, R_{\varphi,s})$. Since $(s^a_i, a, s) \in R^*_{\varphi,s} \setminus R_{\varphi,s} \subseteq R^*_{\varphi,s} \setminus R_{\psi,s}$ for every $1 \le i \le \kappa^a_{\varphi,s}$, we have

$$\mathrm{diff}_a(R^*_{\varphi,s}, R_{\psi,s}) \ge \kappa^a_{\varphi,s} = \mathrm{sd}_a(\varphi) \ge \mathrm{sd}_a(\psi),$$

hence $\lambda^a = \mathrm{sd}_a(\psi)$. On the other hand, since $R_{\psi,s} \subseteq R_{\varphi,s}$ we have

$$\mathrm{sd}_a(\psi) \le \mathrm{sd}_a(\varphi) \le \mathrm{diff}_a(R, R_{\varphi,s}) \le \mathrm{diff}_a(R, R_{\psi,s}),$$

hence $\kappa^a_{\psi,s} = \mathrm{sd}_a(\psi)$, i.e., $\kappa^a_{\psi,s} = \lambda^a$.

Case 2: $\mathrm{sd}_a(\varphi) > \mathrm{diff}_a(R, R_{\varphi,s})$. Then there is exactly one a-transition in $R^*_{\varphi,s}$ for each a-transition in R and vice versa. Since $R_{\psi,s} \subseteq R$ and $R_{\psi,s} \subseteq R^*_{\varphi,s}$ we therefore get $\mathrm{diff}_a(R^*_{\varphi,s}, R_{\psi,s}) = \mathrm{diff}_a(R, R_{\psi,s})$ and hence

$$\kappa^a_{\psi,s} = \min\{\mathrm{sd}_a(\psi), \mathrm{diff}_a(R, R_{\psi,s})\} = \lambda^a.$$

In both cases the same number of states s^a_i together with transitions (s^a_i, a, s) is added for each $a \in \Sigma$. Hence we get $(\mathcal{T}^*_{\varphi,s})^*_{\psi,s} \cong \mathcal{T}^*_{\psi,s}$. ◻

Now we are ready to show the induction step for $\varphi = \psi \vee \chi$. We have

$$\begin{aligned}
& (\mathcal{T}, s) \models \varphi \\
\iff & (\mathcal{T}, s) \models \psi \text{ or } (\mathcal{T}, s) \models \chi \\
\iff & (\mathcal{T}^*_{\psi,s}, s) \models \psi \text{ or } (\mathcal{T}^*_{\chi,s}, s) \models \chi && \text{by induction} \\
\iff & ((\mathcal{T}^*_{\varphi,s})^*_{\psi,s}, s) \models \psi \text{ or } ((\mathcal{T}^*_{\varphi,s})^*_{\chi,s}, s) \models \chi && \text{by Claim 1} \\
\iff & (\mathcal{T}^*_{\varphi,s}, s) \models \psi \text{ or } (\mathcal{T}^*_{\varphi,s}, s) \models \chi && \text{by induction} \\
\iff & (\mathcal{T}^*_{\varphi,s}, s) \models \varphi.
\end{aligned}$$

Claim 2. For $\varphi = \Diamond_a \psi$ and $t \in S$ with $(s, a, t) \in R$ it holds $(\mathcal{T}^*_{\varphi,s})^*_{\psi,t} \cong \mathcal{T}^*_{\psi,t}$.

Proof (of Claim). If there is a π-path from t to v in $\mathcal{T}$ for some $\pi \in P_\psi$, then there is an $a \cdot \pi$-path from s to v and $a \cdot \pi \in P_\varphi$ by definition of P_φ. Hence $S_{\psi,t} \subseteq S_{\varphi,s}$. Analogously we have $R_{\psi,t} \subseteq R_{\varphi,s}$. So $(\mathcal{T}^*_{\varphi,s})_{\psi,t} \cong \mathcal{T}_{\psi,t}$ and it suffices to show that the same number of additional states s_i^b for $b \in \Sigma$ is added to both models. Using the fact that $\mathrm{sd}_b(\varphi) = \mathrm{sd}_b(\psi)$ for every $b \in \Sigma$, the proof is almost the same as for the previous claim (using $R_{\psi,t}$ and $\kappa^b_{\psi,t}$ instead of $R_{\psi,s}$ and $\kappa^a_{\psi,s}$). ◻

Let $\varphi = \Diamond_a \psi$. Since $\varepsilon \in P_\psi$ we have $a \in P_\varphi$. By definition of $\mathcal{T}^*_{\varphi,s}$ there is no a-transition from s to some s_i^b, $b \in \Sigma$. Hence

$$\begin{aligned}
t \in S \wedge (s, a, t) \in R \iff & t \in S_{\varphi,s} \wedge (s, a, t) \in R_{\varphi,s} \\
\iff & t \in S^*_{\varphi,s} \wedge (s, a, t) \in R^*_{\varphi,s}.
\end{aligned} \tag{2}$$

Therefore it holds

$$\begin{aligned}
& (\mathcal{T}, s) \models \varphi \\
\iff & \exists t \in S : (s, a, t) \in R \wedge (\mathcal{T}, t) \models \psi \\
\iff & \exists t \in S : (s, a, t) \in R \wedge (\mathcal{T}^*_{\psi,t}, t) \models \psi && \text{by induction} \\
\iff & \exists t \in S : (s, a, t) \in R \wedge ((\mathcal{T}^*_{\varphi,s})^*_{\psi,t}, t) \models \psi && \text{by Claim 2} \\
\iff & \exists t \in S : (s, a, t) \in R \wedge (\mathcal{T}^*_{\varphi,s}, t) \models \psi && \text{by induction} \\
\iff & \exists t \in S^*_{\varphi,s} : (s, a, t) \in R^*_{\varphi,s} \wedge (\mathcal{T}^*_{\varphi,s}, t) \models \psi && \text{by (2)} \\
\iff & (\mathcal{T}^*_{\varphi,s}, s) \models \varphi.
\end{aligned}$$

Claim 3. Let $\varphi = \Diamond\!\!\!\!-_a \psi$.

1. *For every $t, t' \in S^*_{\varphi,s}$ with $(t, a, t') \in R^*_{\varphi,s}$ there are $u, u' \in S$ with $(u, a, u') \in R$ such that $(\mathcal{T}^*_{\varphi,s} \setminus \{(t, a, t')\})^*_{\psi,s} \cong (\mathcal{T} \setminus \{(u, a, u')\})^*_{\psi,s}$.*
2. *For every $u, u' \in S$ with $(u, a, u') \in R$ there are $t, t' \in S^*_{\varphi,s}$ with $(t, a, t') \in R^*_{\varphi,s}$ such that $(\mathcal{T}^*_{\varphi,s} \setminus \{(t, a, t')\})^*_{\psi,s} \cong (\mathcal{T} \setminus \{(u, a, u')\})^*_{\psi,s}$.*

Proof (of Claim). By definition it holds $P_\varphi = P_\psi$ and therefore $S_{\varphi,s} = S_{\psi,s}$ and $R_{\varphi,s} = R_{\psi,s}$.

1. Case I: If $(t,a,t') \in R_{\varphi,s}$ then also $(t,a,t') \in R$ and we set $u := t$ and $u' := t'$. First we show

$$(\mathcal{T}^*_{\varphi,s} \setminus \{(t,a,t')\})_{\psi,s} \cong (\mathcal{T} \setminus \{(u,a,u')\})_{\psi,s}. \tag{3}$$

Let S_1 and S_2 be the state sets of the left hand side, resp., right hand side and let R_1, R_2 be the corresponding transition relations. It suffices to show $S_1 = S_2$ and $R_1 = R_2$. It holds $v \in S_1$ iff there is a π-path from s to v in $\mathcal{T}^*_{\varphi,s} \setminus \{(t,a,t')\}$ for some $\pi \in P_\psi$, i.e., there is a sequence $\rho = (v_0,a_0,v_1),\dots,(v_{n-1},a_{n-1},v_n)$ with $v_0 = s$, $v_n = v$, $(v_i,a_i,v_{i+1}) \in R^*_{\varphi,s} \setminus \{(t,a,t')\}$ for every $i < n$ and $a_0 \cdots a_{n-1} \in P_\psi$. Since none of the additional states s^b_i, $b \in \Sigma$ has an incoming transition it holds $v_i \in S_{\varphi,s}$ for every $i \leq n$ and $(v_i,a_i,v_{i+1}) \in R_{\varphi,s} \setminus \{(t,a,t')\}$ for every $i < n$. By definition we also have $v_i \in S$ for every $i \leq n$ and $(v_i,a_i,v_{i+1}) \in R \setminus \{(t,a,t')\}$ for every $i < n$. Hence ρ is also a π-path from s to v in $\mathcal{T} \setminus \{(t,a,t')\}$ and therefore $v \in S_2$.

On the other hand, let $v \in S_2$ and ρ be a π-path from s to v in $\mathcal{T} \setminus \{(t,a,t')\}$ for $\pi \in P_\psi$ as above. Then $\rho[0,i]$ is a $\pi[0,i]$-path from s to v_i with $\pi[0,i] \in P_\varphi$ for every $i < n$. Hence $v_i \in S_{\varphi,s}$ for every $i \leq n$, $(v_i,a_i,v_{i+1}) \in R_{\varphi,s} \setminus \{(t,a,t')\}$ for every $i < n$ and ρ is a π-path from s to v in $\mathcal{T}_{\varphi,s} \setminus \{(t,a,t')\} \subseteq \mathcal{T}^*_{\varphi,s} \setminus \{(t,a,t')\}$, i.e., $v \in S_1$ and therefore $S_1 = S_2$. $R_1 = R_2$ is shown analogously.

Next we show that the same number of additional states s^b_i is added to both models in (3), for every $b \in \Sigma$. If $\mathrm{sd}_b(\varphi) > \mathrm{diff}_b(R, R_{\varphi,s})$ then the set of b-transitions in $R^*_{\varphi,s}$ has the same cardinality as the set of b-transitions in R. Since $R_1 = R_2$ we therefore get

$$\mathrm{diff}_b(R^*_{\varphi,s} \setminus \{(t,a,t')\}, R_1) = \mathrm{diff}_b(R \setminus \{(t,a,t')\}, R_2).$$

If $\mathrm{sd}_b(\varphi) \leq \mathrm{diff}_b(R, R_{\varphi,s})$ then the number of additional states s^b_i in $\mathcal{T}^*_{\varphi,s}$ is equal to $\mathrm{sd}_b(\varphi)$ and there are just as many b-transitions in $R^*_{\varphi,s} \setminus R_{\varphi,s}$. Since $R_1 \subseteq R_{\varphi,s}$, $\mathrm{sd}_a(\varphi) = \mathrm{sd}_a(\psi) + 1$ and $\mathrm{sd}_b(\varphi) = \mathrm{sd}_b(\psi)$ for $b \neq a$ it holds

$$\mathrm{diff}_b(R \setminus \{(t,a,t')\}, R_2) \geq \mathrm{diff}_b(R^*_{\varphi,s} \setminus \{(t,a,t')\}, R_1) \geq \mathrm{sd}_b(\varphi) \geq \mathrm{sd}_b(\psi).$$

Therefore the number of additional states s^b_i in both models is equal to $\mathrm{sd}_b(\psi)$.

Case II: If $(t,a,t') = (s^a_i, a, s)$ for some $1 \leq i \leq \kappa^a_{\varphi,s}$ then by definition, there are $u, u' \in R$ with $(u,a,u') \in R \setminus R_{\varphi,s}$. With the notation as before it is easy to see that $S_1 = S_2 = S_{\varphi,s}$ and $R_1 = R_2 = R_{\varphi,s}$, hence (3) is also true for this case. If $b \neq a$ then $\min\{\mathrm{sd}_b(\psi), \mathrm{diff}_b(R^*_{\varphi,s} \setminus \{(s^a_i,a,s)\}, R_1)\} = \min\{\mathrm{sd}_b(\varphi), \mathrm{diff}_b(R^*_{\varphi,s}, R_{\varphi,s})\} = \min\{\mathrm{sd}_b(\varphi), \kappa^b_{\varphi,s}\} = \kappa^b_{\varphi,s}$. On the other hand, $\min\{\mathrm{sd}_b(\psi), \mathrm{diff}_b(R \setminus \{(u,a,u')\}, R_2)\} = \min\{\mathrm{sd}_b(\varphi), \mathrm{diff}_b(R, R_{\varphi,s})\} = \kappa^b_{\varphi,s}$.

For $b = a$ it holds $\min\{\mathrm{sd}_a(\psi), \mathrm{diff}_a(R^*_{\varphi,s} \setminus \{(s^a_i,a,s)\}, R_1)\} = \min\{\mathrm{sd}_a(\varphi) - 1, \kappa^a_{\varphi,s} - 1\} = \kappa^a_{\varphi,s} - 1$ and $\min\{\mathrm{sd}_a(\psi), \mathrm{diff}_a(R \setminus \{(u,a,u')\}, R_2)\} = \min\{\mathrm{sd}_a(\varphi), \mathrm{diff}_a(R, R_{\varphi,s})\} - 1 = \kappa^a_{\varphi,s} - 1$ (note that $(u,a,u') \notin R_2$). So in both cases the number of additional states s^b_i is the same.

2. If $(u,a,u') \in R_{\varphi,s} \subseteq R$ then we set $t := u$ and $t' := u'$ and the proof is exactly the same as for Case I above. Now let $(u,a,u') \in R \setminus R_{\varphi,s}$. Since $\mathrm{sd}_a(\varphi) \geq 1$ we have $\kappa^a_{\varphi,s} \geq 1$ and there is $s^a_1 \in S^*_{\varphi,s} \setminus S_{\varphi,s}$ with $(s^a_1, a, s) \in R^*_{\varphi,s}$. Then we set $t := s^a_1$ and $t' := s$ and repeat the proof of Case II above. $\square$

By using Claim 3 we are able to prove the last induction step. For that let $\varphi = \diamondminus_a \psi$. Then

$$\begin{aligned}
& (\mathcal{T}, s) \models \varphi \\
\iff & \exists u, u' \in S : (u, a, u') \in R \wedge (\mathcal{T} \setminus \{(u, a, u')\}, s) \models \psi \\
\iff & \exists u, u' \in S : (u, a, u') \in R \wedge ((\mathcal{T} \setminus \{(u, a, u')\})^*_{\psi,s}, s) \models \psi && \text{by ind.} \\
\iff & \exists t, t' \in S^*_{\varphi,s} : (t, a, t') \in R^*_{\varphi,s} \wedge ((\mathcal{T}^*_{\varphi,s} \setminus \{(t, a, t')\})^*_{\psi,s}, s) \models \psi && \text{by Cl. 3} \\
\iff & \exists t, t' \in S^*_{\varphi,s} : (t, a, t') \in R^*_{\varphi,s} \wedge (\mathcal{T}^*_{\varphi,s} \setminus \{(t, a, t')\}, s) \models \psi && \text{by ind.} \\
\iff & (\mathcal{T}^*_{\varphi,s}, s) \models \varphi.
\end{aligned}$$

This concludes the proof of the theorem. □

5 Finite Model Property and Satisfiability for PSL

In this section we present five PSL-formulae (α_i, $\beta^a_{k,i}, \gamma_i, \delta_i$ and ζ_i). Together they ensure that a model of an SML-formula φ contains a pruned submodel (relative to φ) such that each two states of the submodel are connected by i-transitions for $1 \leq i \leq \mathrm{sd}(\varphi)$. Further one cannot escape the submodel either by using the standard modalities or by using the sabotage modalities of φ. For technical reasons we additionally use the symbol 0 as a kind of anchor: Deletion of 0-transitions allow us to mark and identify states. Then we are ready to show the main results of the paper: PSL lacks the finite model property and the satisfiability problem for PSL is undecidable.

Let φ be an SML-formula over Σ and let P_φ be as in the last section. We assume that $\Sigma \cap \{0, \ldots, \mathrm{sd}(\varphi)\} = \emptyset$. For a transition system $\mathcal{T} = (S, \Sigma', R, L)$ with $\Sigma \subseteq \Sigma'$ and $s \in S$ let $S_{\varphi,s} \subseteq S$ be defined as before. For a language $A \subseteq \Sigma^*$ the modal operator $\Diamond_A$ is defined by

$$\Diamond_A \psi := \bigvee_{a_1 \cdots a_n \in A} \Diamond_{a_1} \cdots \Diamond_{a_n} \psi.$$

The operator $\Box_A$ is defined analogously. Note that $\Diamond_\emptyset = \bot$ and $\Diamond_{\{\varepsilon\}} \psi = \psi$. In the sequel let $\Sigma_m := \Sigma \mathbin{\dot\cup} \{0, \ldots, m\}$ and $\mathcal{T} = (S, \Sigma_m, R, L)$ be a transition system over Σ_m. The PSL-formula α_i over Σ_m is defined by

$$\alpha_i := \Diamond_0 \top \wedge \diamondminus_0 \Box_0 \bot \wedge \Box_{P_\varphi} \big(\Diamond_0 \top \wedge \diamondminus_i \diamondminus_0 \Box_0 \bot \big).$$

Lemma 12. *If $(\mathcal{T}, s, t) \models \alpha_i$, then $s = t$ and for every $u \in S_{\varphi,s}$ there is $(s, i, u) \in R$ and u has exactly one 0-successor. In particular, $(s, i, s) \in R$.*

Proof. It is easy to see that the first two terms imply $s = t$. If the current position is $[s, s]$, then the last term says that for every $u \in S_{\varphi,s}$ it holds: u has a 0-successor (by $\Diamond_0 \top$) and there is a sabotage path $(s, i, v), (v, 0, w)$ such that u has no 0-successor anymore. Hence it must be $u = v$ and there is only one 0-successor of u. □

For $a \in \Sigma$ the PSL-formula $\beta^a_{k,i}$ over Σ_m is inductively defined by

$$\begin{aligned}
\beta^a_{0,i} &:= \boxdot_i\big(\boxdot_a\bot \vee \Diamond_0\Diamond_{P_\varphi}\Box_0\bot\big),\\
\beta^a_{k+1,i} &:= \Diamond_i\big(\Diamond_0\Box_{P_\varphi}\Diamond_0\top \wedge \Diamond_a\top \wedge \boxdot_{\Sigma\setminus\{a\}}\bot \wedge \boxdot_a(\Diamond_0\Box_0\bot \wedge \beta^a_{k,i})\big).
\end{aligned}$$

Lemma 13. *If $(\mathcal{T}, s, t) \models \alpha_i \wedge \beta^a_{k,i}$ for some $k \in \mathbb{N}$, then there are pairwise different $s^a_1, \ldots, s^a_k \in S$ such that for every $1 \leq j \leq k$:*

1. *$s^a_j \in S \setminus S_{\varphi,s}$, it has a 0-successor and there is $(s, i, s^a_j) \in R$,*
2. *there is $(s^a_j, a, s) \in R$ and if $(s^a_j, a, v) \in R$ for some $v \in S$, then $v = s$,*
3. *s^a_j has no b-successor for $b \in \Sigma$, $b \neq a$.*

On the other hand, if there is $v \in S \setminus S_{\varphi,s}$ with $(s, i, v) \in R$ and $(v, a, s) \in R$, then $v = s^a_j$ for some $1 \leq j \leq k$. In particular, $(\mathcal{T}, s, t) \not\models \beta^a_{l,i}$ for any $l \neq k$.

Proof. By induction on k. By the previous lemma α_i implies $s = t$, so the current position is $[s, s]$. For $k = 0$ assume that there is $v \in S$ with $(s, i, v) \in R$ and $(v, a, s) \in R$. If (s, i, v) is removed and the current position becomes $[s, v]$ then, since v has an a-successor, the second disjunct of $\beta^a_{0,i}$ must be true. This means that there is an outgoing 0-transition of v and, if it is removed, there is a π-path from s to some $u \in S$ for $\pi \in P_\varphi$ such that u has no 0-successor. But by α_i every such u has a 0-successor in the initial model, hence it must be $u = v$ and therefore $v \in S_{\varphi,s}$.

For the induction step we assume that the statement holds for k. If the current position is $[s, s]$ then the first conjunct of $\beta^a_{k+1,i}$ implies that there are $u, v \in S$ and a sabotage path $(s, i, u), (u, 0, v)$ such that every $w \in S_{\varphi,s}$ still has a 0-successor. Hence $u \notin S_{\varphi,s}$. If the current position is $[s, u]$, the second and third conjunct say that u has an a-successor and no b-successor for $b \neq a$. The last term forces that for every a-successor v of u, if the current position is $[s, v]$, then there is $(v, 0, w) \in R$ for some $w \in S$ and, if this transition is removed, s has no 0-successor anymore. But by α_i state s has an initial 0-successor, therefore it must be $v = s$. The current position becomes $[s, s]$ again and by induction $\beta^a_{k,i}$ implies the existence of $s^a_1, \ldots, s^a_k$ with the stated properties. Since the transition (s, i, u) was removed we have $u \neq s^a_j$ for every $1 \leq j \leq k$. Hence we can set $s^a_{k+1} = u$.

Assume that there is $v \in S \setminus S_{\varphi,s}$ and there are $(s, i, v) \in R$ and $(v, a, s) \in R$. If $u \neq v$ then both transitions were not deleted until the current position becomes $[s, s]$ again. By induction, $\beta^a_{k,i}$ implies $v = s^a_j$ for some $1 \leq j \leq k$. $\square$

Let γ_i be the following PSL-formula over Σ_m:

$$\gamma_i := \boxdot_i\big(\Diamond_0\Diamond_{P_\varphi}\Box_0\bot \vee \Diamond_\Sigma\Diamond_0\Box_0\bot\big) \wedge \Box_{P_\varphi}\Box_\Sigma\big(\Diamond_0\top \wedge \Diamond_{P_\varphi}\Diamond_0\Box_0\bot\big).$$

Lemma 14. *Let $k_a \in \mathbb{N}$ for $a \in \Sigma$ such that*

$$(\mathcal{T}, s, t) \models \alpha_i \wedge \bigwedge_{a\in\Sigma} \beta^a_{k_a,i} \wedge \gamma_i.$$

Then for every i-successor v of s, either $v \in S_{\varphi,s}$ or $v = s^a_j$ for some $a \in \Sigma$ and $1 \leq j \leq k_a$ as given in Lemma 13. Further, every Σ-successor of a state in $S_{\varphi,s}$ also belongs to $S_{\varphi,s}$.

Proof. By Lemma 12 the current position is $[s, s]$ and every state $u \in S_{\varphi,s}$ has exactly one 0-successor. By removing (s, i, v) one reaches position $[s, v]$. The first disjunct in the first brackets of γ_i is satisfied if and only if $v \in S_{\varphi,s}$. If $v \in S \setminus S_{\varphi,s}$ then, by the second disjunct, there is a sabotage path $(v, a, w), (w, 0, w')$ for some $a \in \Sigma$ such that s has no 0-successor anymore. Hence $w = s$ and there is $(v, a, s) \in R$. By Lemma 13 we have $v = s^a_j$ for some $1 \leq j \leq k_a$.

Now let $u \in S_{\varphi,s}$ and $v \in S$ with $(u, a, v) \in R$ for some $a \in \Sigma$. By the second conjunct of γ_i, v has a 0-successor and for some $\pi \in P_\varphi$, there is a sabotage π-path from s to some $w \in S_{\varphi,s}$ such that, if the path is extended to some 0-successor of w, then v has no 0-successor anymore. Hence $v = w$ and v belongs to $S_{\varphi,s}$. $\square$

Let δ_i be the following PSL-formula over Σ_m

$$\delta_i := \boxdot_i \Box_i (\Diamond_0 \top \wedge \Diamonddot_i \Diamonddot_0 \Box_0 \bot) \wedge \Box_i \Box_i (\Diamond_0 \top \wedge \Diamonddot_i \Diamonddot_0 \Box_0 \bot).$$

Lemma 15. *If $(\mathcal{T}, s, s) \models \delta_i$, then:*

1. *If $(s, i, u) \in R$ and $(s, i, v) \in R$ for $u \neq v \in S$, then also $(u, i, v) \in R$.*
2. *If $(s, i, u) \in R$ and $(u, i, v) \in R$ for $u, v \in S$, then also $(s, i, v) \in R$.*

Proof. 1. Let $u, v \in S$, $u \neq v$ with $(s, i, u) \in R$ and $(s, i, v) \in R$. By the first conjunct of δ_i, starting from position $[s, s]$ and removing the transition (s, i, u) the current position becomes $[s, u]$. Since (s, i, v) is still available we can reach position $[v, u]$. Then v has a 0-successor and there is a sabotage path $(u, i, w), (w, 0, w')$ such that v has no 0-successor anymore. Hence $v = w$, i.e., there is $(u, i, v) \in R$.

2. Let $u, v \in S$ with $(s, i, u) \in R$ and $(u, i, v) \in R$. By the second conjunct we can reach position $[v, s]$ from the initial position $[s, s]$ and v has a 0-successor. Further there is a sabotage path $(s, i, w), (w, 0, w')$ such that v has no 0-successor anymore. Hence $w = v$, i.e., there is $(s, i, v) \in R$. $\square$

Let ζ_i be the following PSL-formula over Σ_m:

$$\zeta_i := \Box_i \big(\Diamond_0 \top \wedge \big(\Diamonddot_0 \Box_0 \bot \vee \Diamonddot_i (\Diamonddot_0 \Box_0 \bot \wedge \Diamonddot_i \Diamonddot_0 \Box_0 \bot)\big)\big).$$

Lemma 16. *If $(\mathcal{T}, s, s) \models \zeta_i$, then for every $u \in S$ with $(s, i, u) \in R$ there is $(u, i, u) \in R$ and u has exactly one 0-successor.*

Proof. Let the initial position be $[s, s]$ and let $u \in S$ with $(s, i, u) \in R$. If the position becomes $[u, s]$, then u has a 0-successor by the first conjunct of ζ_i. The first disjunct is true if and only if $u = s$ and s has a single 0-successor. In this case we have $(s, i, s) \in R$ by the assumption. If $u \neq s$, then the second disjunct must be satisfied. To satisfy $\Diamonddot_0 \Box_0 \bot$ state u can only have a single 0-successor and one has to remove the transition (s, i, u) such that the current position becomes $[u, u]$. But then one has to use (and remove) an i-transition leading back to u to satisfy the last term, i.e., there must be $(u, i, u) \in R$. $\square$

Now let $m := \mathrm{sd}(\varphi)$ and let $\varphi^{\#}$ be the PSL-formula as defined in Sect. 3. Let $\varphi^{\dagger}$ be the following PSL-formula over Σ_m:

$$\varphi^{\dagger} := \bigwedge_{i=1}^{m} \Big(\alpha_i \wedge \gamma_i \wedge \delta_i \wedge \zeta_i \wedge \Box_i \bigwedge_{n=1}^{m} \blacklozenge_n \blacklozenge_0 \Box_0 \bot \Big) \wedge \bigwedge_{a \in \Sigma} \bigvee_{k=0}^{\mathrm{sd}_a(\varphi)} \bigwedge_{i=1}^{m} \beta^a_{k,i} \wedge \varphi^{\#}.$$

The additional term ensures together with ζ_i that, if $(s, i, u) \in R$ for some $u \in S$ and $1 \leq i \leq m$ then there is also $(s, j, u) \in R$ for every $1 \leq j \leq m$ with $j \neq i$. In particular, the additional states s^a_j due to $\beta^a_{k,i}$ are identical with the ones given by $\beta^a_{k,n}$ for $n \neq i$.

For $\mathcal{T} = (S, \Sigma, R, L)$ and $s \in S$ let $\mathcal{T}^*_{\varphi,s} = (S^*_{\varphi,s}, \Sigma, R^*_{\varphi,s}, L^*_{\varphi,s})$ be defined as in Sect. 4. The transition system $\mathcal{T}^{\dagger}_{\varphi,s}$ is defined by $\mathcal{T}^{\dagger}_{\varphi,s} := (S^*_{\varphi,s}, \Sigma_m, R^{\dagger}, L^*_{\varphi,s})$ where

$$R^{\dagger} := R^*_{\varphi,s} \,\dot{\cup}\, \{(u, 0, u) \mid u \in S^*_{\varphi,s}\} \,\dot{\cup}\, \{(u, i, v) \mid u, v \in S^*_{\varphi,s} \wedge 1 \leq i \leq m\}.$$

Theorem 17. *Let φ be an* SML*-formula over Σ. Then φ is satisfiable iff $\varphi^{\dagger}$ is satisfiable, and φ has a finite model iff $\varphi^{\dagger}$ has a finite model.*

Proof. Let $\mathcal{T} = (S, \Sigma, R, L)$ and $s \in S$ with $(\mathcal{T}, s) \models \varphi$. By Theorem 11 it holds $(\mathcal{T}^*_{\varphi,s}, s) \models \varphi$. Since the symbol 0 does not occur in $\varphi^{\#}$ the same argument as for Lemma 5 shows that $(\mathcal{T}^{\dagger}_{\varphi,s}, s, t) \models \varphi^{\#}$ for any $t \in S$. On the other hand, it is easy to check that $(\mathcal{T}^{\dagger}_{\varphi,s}, s, s)$ satisfies α_i, γ_i, δ_i and ζ_i for every $1 \leq i \leq m$ and that for any $a \in \Sigma$, there is exactly one k with $0 \leq k \leq \mathrm{sd}_a(\varphi)$ (namely $\kappa^a_{\varphi,s}$), such that $\beta^a_{k,i}$ is true for every $1 \leq i \leq m$. Hence $(\mathcal{T}^{\dagger}_{\varphi,s}, s, s) \models \varphi^{\dagger}$, i.e., $\varphi^{\dagger}$ is satisfiable and if $\mathcal{T}$ is a finite model of φ, then $\mathcal{T}^{\dagger}_{\varphi,s}$ is a finite model of $\varphi^{\dagger}$.

For the converse let $\mathcal{T} = (S, \Sigma_m, R, L)$ and $s, t \in S$ such that $(\mathcal{T}, s, t) \models \varphi^{\dagger}$. By Lemma 12 it holds $s = t$. By Lemma 13 there is exactly one k_a for every $a \in \Sigma$ with $0 \leq k_a \leq \mathrm{sd}_a(\varphi)$ such that $\beta^a_{k_a,i}$ is satisfied. Let $S_{\varphi,s} \subseteq S$ be as before and let $S' \subseteq S$ be defined by

$$S' := S_{\varphi,s} \,\dot{\cup}\, \{s^a_j \mid a \in \Sigma \wedge 1 \leq j \leq k_a\},$$

where the s^a_j's in $S \setminus S_{\varphi,s}$ are according to Lemma 13. Note that we have $(s, i, s^a_j) \in R$ for every $1 \leq i \leq m$ by the additional term in $\varphi^{\dagger}$. Each s^a_j has a single outgoing Σ-transition which is labeled by a and leads to s. By Lemmata 12, 13, 16, there is $(s, i, u) \in R$ for every $u \in S'$ and u has exactly one 0-successor. Since only the existence of 0-successors is used in all subformulae, but none of these transitions is actually traversed, we can assume that all 0-transitions occur as loops, i.e., there is $(u, 0, u) \in R$ for every $u \in S'$ and $(u, 0, v) \notin R$ for $u, v \in S', u \neq v$. Further, there is $(u, i, v) \in R$ for every $u, v \in S'$ (by Lemma 15, if $u \neq v$ and by Lemma 16, if $u = v$). Let $u \in S'$ and $v \in S$ with $(u, i, v) \in R$. Since there is $(s, i, u) \in R$, there is also $(s, i, v) \in R$ by Lemma 15. By Lemma 14 it follows $v \in S'$, i.e., one cannot escape S' by using i-transitions. On the other hand, again by Lemma 14, one cannot escape $S_{\varphi,s}$ by using Σ-transitions. In other words, using any modality in $\varphi^{\dagger}$ – either a standard or a sabotage one –

one stays in S'. It is easy to see that we can therefore restrict $\mathcal{T}$ to the states in S', i.e., for the transition system $\mathcal{T}' := (S', \Sigma_m, R \cap S' \times \Sigma_m \times S', L|_{S'})$ it also holds $(\mathcal{T}', s, s) \models \varphi^\dagger$. In particular, $(\mathcal{T}', s, s)$ is a model of $\varphi^\#$ with i-transitions between any two states. Let $\mathcal{T}''$ be the restriction of $\mathcal{T}'$ to the alphabet Σ. Since the symbol 0 does not occur in $\varphi^\#$ and by the same argument as for Lemma 5 we get $(\mathcal{T}'', s) \models \varphi$, i.e., φ is satisfiable. Further, if $\mathcal{T}$ is a finite model of $\varphi^\dagger$, then $\mathcal{T}''$ is a finite model of φ. $\square$

Now we are ready to transfer the results on SML to PSL. By using the reduction $\varphi \mapsto \varphi^\dagger$ together with Theorem 2 and Theorem 3 we get

Corollary 18. *The logic* PSL *does not have the finite model property. The decision problems* Satisfiability*,* Finite Satisfiability*, and* Infinity Axiom *for* PSL *are undecidable.* $\square$

6 Conclusion

We have considered the path sabotage logic PSL which is a balanced version of SML. Both logics are extensions of modal logic that are capable of describing elementary changes of structures. We have shown that the model checking complexity for the logic PSL with a localized sabotage modality is as hard as for SML that has a global 'edge-deleting' modality. Also the satisfiability problem stays undecidable. In fact, from the viewpoint of complexity, both logics much more resemble first-order logic than modal logic, except for a linear formula and a polynomial program complexity.

There are other restrictions to the global power of the sabotage operator, for example the localized version of SML where only those edges can be deleted that start at the current position within the system. Interpreting the modalities as movements of the agents 'runner' and 'saboteur' in a crumbling network, this localized sabotage logic corresponds to the situation that the saboteur can only block adjacent nodes and that the runner gives the saboteur a 'pickaback' while moving in the network. An argument (to be presented elsewhere) which resembles the proofs above shows that the complexities stay the same: Uniform model checking is PSPACE-complete and the satisfiability problem is undecidable.

References

1. J. v. Benthem. An essay on sabotage and obstruction. In D. Hutter and S. Werner, editors, *Festschrift in Honour of Prof. Jörg Siekmann*, LNAI. Springer, 2002.
2. R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning about Knowledge.* MIT Press, 1995.
3. Ch. Löding, Ph. Rohde. Solving the sabotage game is PSPACE-hard. In Proceedings of MFCS 2003. Vol. 2747 of LNCS, Springer (2003), pp. 531–540
4. Ch. Löding, Ph. Rohde. Model checking and satisfiability for sabotage modal logic. In Proceedings of FSTTCS 2003. Vol. 2914 of LNCS, Springer (2003), pp. 302–313