A Note on an Extension of PDL

Stefan Göller

Institut for Computer Science
University of Leipzig, Germany

Dirk Nowotka

Institute for Formal Methods in Computer Science (FMI)
University of Stuttgart, Germany

Abstract

Recently visibly pushdown automata have been extended to so called k-phase multi-stack visibly pushdown automata (k-MVPAs). On the occasion of introducing k-MVPAs, it has been asked whether the extension of Propositional Dynamic Logic with k-MVPAs still leads to a decidable logic. This question is answered negatively here.

Key words: Propositional Dynamic Logic, Visibly Pushdown Automata, Multi-Stack Visibly Pushdown Automata, Decidability, Satisfiability

1 Introduction

Propositional Dynamic Logic (PDL) is a modal logic introduced by Fischer and Ladner [1] which allows to reason about regular programs. In PDL, there are two syntactic entities: formulas, built from boolean and modal operators and interpreted as sets of worlds of a Kripke structure; and programs, built from the operators test, union, composition, and Kleene star and interpreted as binary relations in a Kripke structure. Hence, the occuring programs can be seen as a regular language over an alphabet that consists of tests and atomic programs. However, the mere usage of regular programs limits the expressiveness of PDL as for example witnessed by the set of executions of well-matched

Email addresses: goeller@informatik.uni-leipzig.de (Stefan Göller), nowotka@fmi.uni-stuttgart.de (Dirk Nowotka).

calls and returns of a recursive procedure, cf. [2]. Therefore, non-regular extensions of PDL have been studied quite extensively [2–5]. An extension of PDL by a class $\mathcal{L}$ of languages means that in addition to regular languages also languages in $\mathcal{L}$ may occur in modalities of formulas. One interesting result on PDL extensions, among many others as summarized in [2], is the extension of PDL with visibly pushdown languages [6] which are the languages recognized by visibly pushdown automata (VPA). A VPA is a pushdown automaton, where the stack operation is determined by the input in the following way; the alphabet is partitioned into letters that prompt a push, internal, or pop action, respectively. Löding, Lutz, and Serre [4] showed that satisfiability of a PDL extension with VPL is complete for deterministic doubly exponential time. Note that for this result, every visibly pushdown language occuring in a formula must be over the same partition of the alphabet.

Recently, k-phase multi-stack visibly pushdown automata (k-MVPAs), a natural extension of VPAs, have been introduced in [7]. A k-MVPA is an automaton equipped with n stacks where, again, the actions on the stacks are determined by the input, more precisely, every input symbol specifies on which stack a push or pop operation or whether an internal operation is done. Moreover, a k-MVPA is restricted to accept only words that can be obtained by concatenating at most k *phases*, where a phase is a sequence of input symbols that invoke pop actions from at most one stack. Note that k-MVPAs with one stack coincide with VPAs. The language class k-MVPL that is described by k-MVPAs has various effective closure properties and a decidable nonemptiness problem. Therefore it is an interesting question to ask if the corresponding extension of PDL is still decidable. This question was raised in [7] and is answered negatively in this note. We prove Σ_1^1-completeness for this PDL extension. A Σ_1^1 lower bound already holds, if we extend PDL with two fixed languages each accepted by some 2-MVPA over two stacks, namely $\{(a_1b_2)^n(a_2b_1)^n \mid n \geq 1\}$ and $\{(a_2b_1)^n(a_1b_2)^n \mid n \geq 1\}$. For this, we give an easy reduction of PDL with the two languages $\{a^nb^n \mid n \geq 1\}$ and $\{b^na^n \mid n \geq 1\}$, proven to be Σ_1^1 hard in [3], to the latter PDL extension.

2 Propositional Dynamic Logic Extensions

Fix some countable set $\mathbb{P}$ of *atomic propositions*, some finite alphabet Σ and a class of languages $\mathcal{L} \subseteq 2^{\Sigma^*}$. The *set of formulas* Φ and the set of *tests* Tests of the logic PDL+$\mathcal{L}$ are the smallest sets that satisfy the following conditions: (i) if $p \in \mathbb{P}$, then $p \in \Phi$, (ii) if $\varphi_1, \varphi_2 \in \Phi$, then $\varphi_1 \vee \varphi_2, \neg\varphi_1 \in \Phi$, (iii) if $\varphi \in \Phi$, then $\varphi? \in$ Tests, and (iv) if $\varphi \in \Phi$ and $\Psi \subset$ Tests is finite, then $\langle\chi\rangle\varphi \in \Phi$, where χ is a regular expression over $\Sigma \cup \Psi$ or $\chi \in \mathcal{L}$.

A *Kripke structure* is a tuple $K = (X, \{\rightarrow_a \mid a \in \Sigma\}, \{X_p \mid p \in \mathbb{P}\})$, where X is a set of *worlds*, $\rightarrow_a \subseteq X \times X$ is a binary relation for each $a \in \Sigma$, and $X_p \subseteq X$ is a unary relation for each $p \in \mathbb{P}$. For each $\varphi \in \Phi$ and for each $w \in (\Sigma \cup \mathrm{Tests})^*$, define the binary relation $[\![w]\!]_K \subseteq X \times X$ and the set $[\![\varphi]\!]_K \subseteq X$ via mutual induction as follows:

- $[\![\varepsilon]\!]_K = \{(x,x) \mid x \in X\}$,
- if $\varphi? \in \mathrm{Tests}$, then $[\![\varphi?]\!]_K = \{(x,x) \mid x \in X \wedge x \in [\![\varphi]\!]_K\}$,
- if $a \in \Sigma$, then $[\![a]\!]_K = \rightarrow_a$,
- if $w \in (\Sigma \cup \mathrm{Tests})^*$ and $\tau \in \Sigma \cup \mathrm{Tests}$, then $[\![w\tau]\!]_K = [\![w]\!]_K \circ [\![\tau]\!]_K$,
- if $p \in \mathbb{P}$, then $[\![p]\!]_K = X_p$,
- $[\![\varphi_1 \vee \varphi_2]\!]_K = [\![\varphi_1]\!]_K \cup [\![\varphi_2]\!]_K$,
- $[\![\neg\varphi]\!]_K = X \setminus [\![\varphi]\!]_K$,
- $[\![\langle\chi\rangle\varphi]\!]_K = \{x \in X \mid \exists y \in X\ \exists w \in L(\chi) : (x,y) \in [\![w]\!]_K \wedge y \in [\![\varphi]\!]_K\}$.

We say that K is a *model* for φ, if $x \in [\![\varphi]\!]_K$ for some world x of K. We say that a formula φ is *satisfiable*, if there exists a model for φ. The *satisfiability problem* asks, given a formula φ, whether φ is satisfiable.

3 Σ_1^1-Completeness of PDL+k-MVPL

It is not hard to see that satisfiability of PDL+k-MVPL is in Σ_1^1. Firstly, we can easily adapt the proof of Proposition 9.4 in [2] and show that every satisfiable PDL+k-MVPL formula φ has a countable tree model. Secondly, we can write down an existential second-order formula over $\mathbb{N}$ that is valid if and only if φ is satisfiable.

Before we give the matching Σ_1^1 lower bound let us, for finite words u and v, define the language $u^N v^N := \{u^n v^n \mid n \geq 1\}$. In the following, fix the language class $\mathcal{L}_0 = \{(a_1b_2)^N(a_2b_1)^N, (a_2b_1)^N(a_1b_2)^N\}$. The following Proposition is obvious.

Proposition 1 *There exist* 2*-MVPAs* M_1, M_2 *over a common* 2*-stack alphabet such that* $L(M_1) = (a_1b_2)^N(a_2b_1)^N$ *and* $L(M_2) = (a_2b_1)^N(a_1b_2)^N$.

For the lower bound, we prove that already satisfiability for PDL+$\mathcal{L}_0$ is Σ_1^1-hard. For this, we use the following result.

Theorem 2 ([3]) *Satisfiability of PDL+*$\{a^N b^N, b^N a^N\}$ *is* Σ_1^1*-hard.*

Hence by Fact 1 and Theorem 2 it remains to give a satisfiability preserving translation from PDL+$\{a^N b^N, b^N a^N\}$ to PDL+$\mathcal{L}_0$. The translation is straightforward. Define a homomorphism $h : \{a,b\}^* \rightarrow \{a_1,a_2,b_1,b_2\}^*$ as follows, $h(a) = a_1b_2$ and $h(b) = a_2b_1$. For a PDL+$\{a^N b^N, b^N a^N\}$ formula φ, let the PDL+$\mathcal{L}_0$ formula φ' emerge from φ by replacing each occurence of a by $h(a)$

and each occurence of b by $h(b)$. Finally, it suffices to prove the following lemma.

Lemma 3 *The formula φ is satisfiable if and only if φ' is satisfiable.*

PROOF. "Only-if": Assume φ is satisfiable. Let $K = (X, \{\rightarrow_a, \rightarrow_b\}, \{X_p \mid p \in \mathbb{P}\})$ be a model of φ, i.e. $x_0 \in [\![\varphi]\!]_K$ for some $x_0 \in X$. Let K' be the Kripke structure that is obtained from K by (i) replacing each transition $x \rightarrow_a y$ by a sequence of two transitions $x \rightarrow_{a_1} z \rightarrow_{b_2} y$ for some fresh world z in K', (ii) replacing each transition $x \rightarrow_b y$ by a sequence of two transitions $x \rightarrow_{a_2} z \rightarrow_{b_1} y$ for some fresh world z in K', and (iii) keeping X_p unchanged for each $p \in \mathbb{P}$. Now for each $x, y \in X$ and for each $w \in \{a, b\}^*$ we have $(x, y) \in [\![w]\!]_K$ if and only if $(x, y) \in [\![h(w)]\!]_{K'}$. By an induction on the structure of φ, one can prove that $x_0 \in [\![\varphi']\!]_{K'}$. Thus, K' is a model for φ'.

"If": Assume φ' is satisfiable. Let $K' = (X, \{\rightarrow_{a_i}, \rightarrow_{b_i} \mid i = 1, 2\}, \{X_p \mid p \in \mathbb{P}\})$ be a model of φ', i.e. $x_0 \in [\![\varphi']\!]_{K'}$ for some $x_0 \in X$. Now define the Kripke strucutre $K = (X, \{\rightarrow_a, \rightarrow_b\}, \{X_p \mid p \in \mathbb{P}\})$ where $\rightarrow_a = \{(x, y) \in X \times X \mid \exists z : x \rightarrow_{a_1} z \rightarrow_{b_2} y\}$ and $\rightarrow_b = \{(x, y) \in X \times X \mid \exists z : x \rightarrow_{a_2} z \rightarrow_{b_1} y\}$. As above for each $x, y \in X$ and for each $w \in \{a, b\}^*$ we have $(x, y) \in [\![w]\!]_K$ if and only if $(x, y) \in [\![h(w)]\!]_{K'}$. Again, by an induction on the structure of φ, one can prove that $x_0 \in [\![\varphi]\!]_K$. Thus, K is a model for φ.

References

[1] M. J. Fischer, R. E. Ladner, Propositional dynamic logic of regular programs, J. Comput. Syst. Sci. 18 (2) (1979) 194–211.

[2] D. Harel, D. Kozen, J. Tiuryn, Dynamic Logic, Foundations of computing, The MIT Press, 2000.

[3] D. Harel, A. Pnueli, J. Stavi, Propositional Dynamic Logic of Nonregular Programs, J. Comput. System Sci. 26 (2) (1983) 222–243.

[4] C. Löding, C. Lutz, O. Serre, Propositional dynamic logic with recursive programs, J. Log. Algebr. Program. 73 (1–2) (2007) 51–69.

[5] T. Koren, A. Pnueli, There exist decidable context-free propositional dynamic logics, in: Proceedings of the Carnegie Mellon Workshop on Logic of Programs, Springer-Verlag, London, UK, 1984, pp. 290–312.

[6] R. Alur, P. Madhusudan, Visibly pushdown languages, in: Proceedings of the 36th Annual ACM Symposium on Theory of Computing, ACM, New York, 2004, pp. 202–211 (electronic).

[7] S. L. Torre, P. Madhusudan, G. Parlato, A robust class of context-sensitive languages, in: Proceedings of the 22nd IEEE Symposium on Logic in Computer Science, IEEE, 2007, pp. 161–170.