PDL over Accelerated Labeled Transition Systems

Taolue Chen
CWI
PO Box 94079
1090 GB Amsterdam, NL
chen@cwi.nl

Jaco van de Pol
University of Twente
PO Box 217
7500 AE Enschede, NL
vdpol@cs.utwente.nl

Yanjing Wang
CWI
PO Box 94079
1090 GB Amsterdam, NL
y.wang@cwi.nl

Abstract

We present a thorough study of Propositional Dynamic Logic *over a variation of labeled transition systems, called* accelerated labelled transition systems*, which are transition systems labeled with* regular expressions *over action labels. We study the model checking and satisfiability decision problems. Through a notion of regular expression rewriting, we reduce these two problems to the corresponding ones of PDL in the traditional semantics (w.r.t. LTS). As for the complexity, both of problems are proved to be* EXPSPACE*-complete. Moreover, the program complexity of model checking problem turns out to be* NLOGSPACE*-complete. Furthermore, we provide an axiomatization for PDL which involves Kleene Algebra as an Oracle. The soundness and completeness are shown.*

1 Introduction

Automatic verification techniques, such as model checking [4], normally require the exploration of a labeled transition system (LTS) corresponding to a formal specification. These techniques are quite limited by the size of the state space, which may be too large or even infinite. *Abstraction* is being widely used to reduce the complexity of the analyzed systems.

We express system properties in Propositional Dynamic Logic (PDL[5]). This was introduced by Fischer and Landner in the late 70s as a formalism for reasoning on programs. Its main operators state that some property holds after all or some executions matching a given regular expression.

In order to preserve universal and existential properties one typically uses three-valued logic [1] on modal labeled transition systems (MLTS) [14, 6]. Universal properties (safety) are checked on an overapproximation (may transitions), while existential properties (liveness) are checked on an underapproximation (must transitions). This works fine for *safety* properties, but the verification of *liveness* properties is problematic. The problem comes from the lack of guaranteed (required) behaviors, due to the nondeterminism introduced by abstraction.

Accelerated Labeled Transition Systems. To deal with this problem, Valero Espada and the second author proposed *accelerated modal LTS* (AMLTS), a new formalism to represent abstractions [16]. They enhance Modal-LTSs by labeling must-transitions with *sequences of actions*. These so-called *accelerated* transitions capture the idea that a state can be reached from another state by some finite computation. In the current paper, we study accelerated transitions only, and talk about Accelerated Labelled Transition Systems (ALTS).

This extension captures abstract systems more accurately and therefore infers stronger *liveness* properties. As an example, abstracting a count-down process could involve states $zero$ and pos. There would be may-transitions from pos to pos and to $zero$ (both labeled by a dec-action), but no must-transitions at all. However, one could introduce an accelerated must-transition from pos to $zero$ (labeled by dec^+).

Main contributions. Usually, PDL is interpreted over an LTS, but in [16] PDL is interpreted over an ALTS. We will see that this makes a big difference. Developing a model checking algorithm is of utmost importance. Moreover, for an in-depth understanding of the logic, axiomatization and satisfiability checking are two central questions. We explore all of these problems.

A model checking algorithm should check whether a PDL formula holds for an ALTS. In [16], an algorithm with high complexity is provided, showing decidability of the model checking problem. It is quite different from the usual PDL model checking algorithm (see, e.g. [11]). A hard problem left open in [16] is the precise complexity and optimality of the algorithm.

In Section 3, we provide a model checking algorithm for PDL on ALTS, by exploiting the notion of regular expression rewriting studied extensively in [3]. The complexity can be easily analyzed, namely, in EXPSPACE. Further-

978-0-7695-3249-3/08 $25.00 © 2008 IEEE
DOI 10.1109/TASE.2008.42

more, we prove an EXPSPACE lower bound for the model checking problem. This result solves an open problem left in [16] and establishes a strong link between model checking PDL over ALTS and regular expression rewriting. In Section 4, we provide an axiomatization of PDL on ALTS, which employs *Kleene Algebra* [10] as an oracle. The soundness and completeness are shown. This result shows very clearly the differences with traditional PDL on LTS. Furthermore, in Section 5, we study the satisfiability decision problem. By, again, resorting to the notion of regular expression rewriting [3], we reduce this problem to the satisfiability of PDL in the traditional semantics (over LTS) and show that satisfiability of PDL over ALTS is also EXPSPACE-complete.

Related work. We mention some related work: finite-state automata that allow more complex transition labels recently received a resurgence of attention. These include *generalized automata* [7] (a.k.a. string or lazy automata) with strings (or blocks) as transition labels rather than merely characters or the null string and *expression automata* [9], finite-state automata whose transition labels are regular expressions over the input alphabet. These share the same idea as our accelerated LTS. However, they mainly studied these extended automata from the automata and language perspectives, in particular, the determinism and minimization problems are explored there. In logic, [15] studies μ-calculus with regular expressions in the modalities. It is shown that in this case, regular expressions in formulae can be easily eliminated by the fixpoint construction. [13] introduces the notion of regular linear temporal logic, which is a logic that generalizes linear temporal logic with the ability to use regular expressions arbitrarily as sub-expressions. The expressiveness and satisfiability of this logic are investigated there. These works are orthogonal to regular expressions in the LTS, which is the main focus of the current paper.

2 Preliminaries

2.1 Accelerated Labelled Transition Systems

Given an alphabet Σ, *regular expressions* over Σ are of the form

$$\alpha ::= a \mid \alpha + \alpha \mid \alpha \cdot \alpha \mid \alpha^*$$

where $a \in \Sigma$. We write Σ^* as the set of words over Σ.

The interpretation of regular expression α, namely, the regular language of α, is denoted by $\mathcal{L}(\alpha)$:

$$\begin{array}{lcl} \mathcal{L}(a) & = & \{a\} \\ \mathcal{L}(\alpha_1 + \alpha_2) & = & \mathcal{L}(\alpha_1) \cup \mathcal{L}(\alpha_2) \\ \mathcal{L}(\alpha^*) & = & \mathcal{L}(\alpha)^* \\ \mathcal{L}(\alpha_1 \cdot \alpha_2) & = & \{w_1 \cdot w_2 \mid w_1 \in \mathcal{L}(\alpha_1), w_2 \in \mathcal{L}(\alpha_2)\} \end{array}$$

Definition 1 [Accelerated Labelled Transition System] An *Accelerated Labelled Transition System* (ALTS) is a tuple $\mathcal{M} = (S, Act, \rightarrow, V)$ where

- S is a non-empty set of states;
- Act is a non-empty set of atomic action labels;
- $\rightarrow$ is a possibly infinite set of *accelerated* transitions of the form $s \xrightarrow{\sigma} s'$ with $s, s' \in S$, and σ being a regular expression over alphabet Act.
- V is the valuation function: $V : S \rightarrow 2^{\Phi}$ where Φ is a set of atomic propositions.

Following the tradition in modal logic, we shall call $\mathcal{F} = (S, Act, \rightarrow)$ an ALTS *frame*.

A *Labeled Transition System* (LTS) is an ALTS with the constraint that every transition is labeled by a *single atomic action*.

2.2 Propositional Dynamic Logic

Propositional Dynamic Logic (PDL) is a branching-time logic, in the style of Hennessy-Milner Logic with regular expressions:

$$\varphi ::= \top \mid p \mid \varphi \wedge \varphi \mid \neg\varphi \mid \langle\alpha\rangle\varphi$$

where, p is an atomic proposition and α is a regular expression over some alphabet Σ. When Σ is not fixed, we use PDL_Σ to denote the PDL language based on Σ.

As usual, we define $\bot$, $\phi \vee \psi$, $\phi \rightarrow \psi$ and $[\beta]\phi$ as the abbreviations of $\neg\top$, $\neg(\neg\phi \wedge \neg\psi)$, $\neg\phi \vee \psi$ and $\neg\langle\beta\rangle\neg\phi$ respectively.

As the semantics, $\langle\alpha\rangle\varphi$ holds in a state in which there exists at least one α sequence to a state satisfying φ while $[\alpha]\varphi$ holds in a state in which all continuations by sequences matching α end in a state satisfying φ. We define the satisfiability relation $\models$ between a pointed model $\mathcal{M}, s$ and a PDL formula φ as follows:

$$\begin{array}{lcl} \mathcal{M}, s \models \top & \iff & \text{always} \\ \mathcal{M}, s \models p & \iff & s \in V(p) \\ \mathcal{M}, s \models \neg\phi & \iff & \mathcal{M}, s \not\models \phi \\ \mathcal{M}, s \models \phi \wedge \psi & \iff & \mathcal{M}, s \models \phi \text{ and } \mathcal{M}, s \models \psi \\ \mathcal{M}, s \models \langle\beta\rangle\phi & \iff & \text{there exists a path} \\ & & s = s_0 \xrightarrow{\sigma_1} s_1 \xrightarrow{\sigma_2} \cdots \xrightarrow{\sigma_n} s_n \\ & & \text{in } \mathcal{M} \text{ such that } \mathcal{M}, s_n \models \phi \text{ and} \\ & & \mathcal{L}(\sigma_1\sigma_1 \ldots \sigma_n) \subseteq \mathcal{L}(\beta) \end{array}$$

To illustrate the semantics, we present two simple examples:

Example 1

$\mathcal{M}, s \vDash \langle a+b\rangle\top$ $\quad$ $\mathcal{M}, t \vDash \langle a+b\rangle\top$

$\mathcal{M}, s \nvDash \langle a\rangle\top$ $\quad$ $\mathcal{M}, t \vDash \langle a\rangle\top, \langle b\rangle\top$

Recall that for the standard PDL semantics on LTS (called $\mathcal{G}$) with action set Act, the satisfiability relation $\Vdash$ for modality cases are defined as:

- $\mathcal{G}, s \Vdash \langle\beta\rangle\phi \iff \exists$ path $s_0 \xrightarrow{e_0} s_1 \xrightarrow{e_1} \cdots \xrightarrow{e_{n-1}} s_n$ in $\mathcal{M}$ such that $e_1 \dots e_n \in \mathcal{L}(\beta)$ and $\mathcal{M}, s_n \Vdash \phi$.
- $[\beta]\phi$ is dual to $\langle\beta\rangle\phi$.

2.3 Regular Expression Rewriting

The notion of *regular expression rewriting* is introduced in [3], and turns out to play an essential role in solving model checking and satisfiability checking problems. The following exposition is taken from [3].

Given a regular expression β and a finite set $\mathcal{E} = \{\alpha_1, \dots, \alpha_k\}$ of regular expressions over an alphabet Σ, re-express, if possible, β by a suitable combination of $\alpha_1, \dots, \alpha_k$. We assume that associated with $\mathcal{E}$ we always have an alphabet $\Sigma_\mathcal{E}$ containing exactly one unique symbol e_α for each α in $\mathcal{E}$, and we use $re(e)$ to denote the regular expression associated with the symbol $e \in \Sigma_\mathcal{E}$. Given any language L over $\Sigma_\mathcal{E}$, we denote by $exp_\Sigma(L)$ the expansion of L w.r.t. $\mathcal{E}$, i.e., the language over Σ defined as follows

$$exp_\Sigma(L) = \bigcup_{e_1 \cdots e_n \in L} \{w_1 \cdots w_n \mid w_i \in \mathcal{L}(re(e_i))\}$$

where $\mathcal{L}(\alpha)$ is the language defined by the regular expression α. Thus, $exp_\Sigma(L)$ denotes all the words obtained from a word $e_1, \cdots e_n \in L$ by substituting for each e_i all words of the regular languages associated with e_i. Given a $\Sigma_\mathcal{E}$-word w, $exp_\Sigma(\{w\})$ is simply called the *expansion* of w.

Definition 2 Let α be a regular expression over the alphabet $\Sigma_\mathcal{E} = \{e_0, e_1, \dots e_n\}$. We say α is a rewriting of β (a regular expression over Σ) w.r.t $\mathcal{E}$ if $exp_\Sigma(\mathcal{L}(\alpha)) \subseteq \mathcal{L}(\beta)$. α is called a $\Sigma_\mathcal{E}-$*maximal rewriting* if for any other rewriting α' of β w.r.t. $\Sigma_\mathcal{E}$: $\mathcal{L}(\alpha') \subseteq \mathcal{L}(\alpha)$ (thus $exp_\Sigma(\alpha') \subseteq exp_\Sigma(\alpha)$). We say that a rewriting α is *empty* if $\mathcal{L}(\alpha) = \emptyset$.

In [3], the problem of finding a maximal rewriting is shown to be EXPSPACE-complete.

3 Model Checking

In this section, we tackle the model checking problem. At the first sight, one might think this is a very simple problem: an immediate idea might be first to transform an ALTS into LTS by expanding, then run traditional model checking algorithm. However, this does *not* work, at least not in a naive way. Let us look at Example 1, left figure. Suppose one wants to check $\langle a\rangle\top$ which is *false* and following this idea, one can obtain a LTS in the right figure. However, the result will be *true*. The other naive idea is to "merge" the transitions in the ALTS such that for any two states, there is only one accelerated transition between them. This does *not* work well either: Suppose the considered ALTS is Example 1, right figure, and one wants to check $\langle a\rangle\top$, which is *true*. However, after the transformation, the left figure is obtained and the result would be *false*. These two examples suggest that the model checking can not be performed in a very simple way.

3.1 Algorithm

We now present the correct algorithm, where the idea is to reduce the model checking problem of PDL over ALTS to the one over LTS in a more sophisticated manner. Here, as said, the notion of regular expression rewriting is exploited.

Notation. Given a set of regular expressions $\{\beta, \alpha_1, \dots \alpha_n\} \subseteq \Sigma^*$, let $\mathcal{E} = \{\alpha_1, \dots, \alpha_n\}$, $\hat{\beta}_\mathcal{E}$ be the *maximal* $\Sigma_\mathcal{E}-$*rewriting* of β. Note that $\hat{\beta}_\mathcal{E}$ is a regular expression over $\Sigma_\mathcal{E} = \{e_\alpha \mid \alpha \in \mathcal{E}\}$ and can be computed by an algorithm in [3].

Definition 3 Given an ALTS $\mathcal{M} = (S, Act, \to, V)$, let

$$\langle\rangle_\mathcal{M} = \{\sigma \mid \sigma \in Act^* \text{ and } \sigma \text{ appears in some transition of } \mathcal{M}\}$$

We define $\ulcorner\mathcal{M}\urcorner$ as $(S, \{e_\alpha \mid \alpha \in \langle\rangle_\mathcal{M}\}, \to', V)$ where $s \xrightarrow{e_\alpha}' s'$ iff $s \xrightarrow{\alpha} s'$.

Definition 4 [Rewriting w.r.t an ALTS] Given a ALTS $\mathcal{M}$ and a PDL formula ϕ, $\mathfrak{R}_\mathcal{M}(\phi)$ is the rewriting of ϕ in language $\text{PDL}_{\Sigma_{\langle\rangle_\mathcal{M}}}$ defined by: [1]

- $\mathfrak{R}_\mathcal{M}(p) = p$ where p an atomic proposition;
- $\mathfrak{R}_\mathcal{M}(\neg\psi) = \neg\mathfrak{R}_\mathcal{M}(\psi)$;
- $\mathfrak{R}_\mathcal{M}(\psi_1 \wedge \psi_2) = \mathfrak{R}_\mathcal{M}(\psi_1) \wedge \mathfrak{R}_\mathcal{M}(\psi_2)$;
- $\mathfrak{R}_\mathcal{M}(\langle\alpha\rangle(\psi)) = \langle\hat{\alpha}_{\langle\rangle_\mathcal{M}}\rangle\mathfrak{R}_\mathcal{M}(\psi)$.

Theorem 1 For any pointed ALTS $\mathcal{M}, s$ and any PDL formula ϕ,

$$\mathcal{M}, s \vDash \phi \iff \ulcorner\mathcal{M}\urcorner, s \Vdash \mathfrak{R}_\mathcal{M}(\phi).$$

[1] Since rewriting might introduce ϵ (the language only containing empty word) and δ (empty language), for technical convenience, we add ϵ and δ into the language of PDL programs when traditional PDL semantics is concerned. It is not hard to see they are auxiliary and can be eliminated in the standard semantics of PDL since $[\epsilon]\phi \leftrightarrow \phi$ and $[\delta]\phi$ are valid.

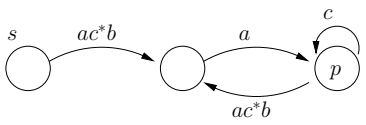

Figure 1. Accelerated LTS

Proof: By induction on the structure of ϕ. The only interesting case is $\phi = \langle\alpha\rangle\psi$.

$\Rightarrow$) Suppose $\mathcal{M}, s \models \langle\alpha\rangle\psi$ then there exists some t in $\mathcal{M}$ such that $s \xrightarrow{\sigma_1} \cdots \xrightarrow{\sigma_n} t$ in $\mathcal{M}$ and $\mathcal{L}(\sigma_1 \cdots \sigma_n) \subseteq \mathcal{L}(\alpha)$. Since $\hat{\alpha}_{\langle\rangle_\mathcal{M}}$ is the maximal $\Sigma_{\langle\rangle_\mathcal{M}}$ rewriting of ϕ and $\{\sigma_1, \ldots, \sigma_n\} \subseteq \langle\rangle_\mathcal{M}$, $\mathcal{L}(e_{\sigma_1} \cdots e_{\sigma_n}) \subseteq \mathcal{L}(\hat{\alpha}_{\langle\rangle_\mathcal{M}})$. It follows that $e_{\sigma_1} \cdots e_{\sigma_n} \in \mathcal{L}(\hat{\alpha}_{\langle\rangle_\mathcal{M}})$. By induction hypothesis, $\ulcorner\mathcal{M}\urcorner, t \Vdash \mathfrak{R}_\mathcal{M}(\psi)$ and thus $\ulcorner\mathcal{M}\urcorner, s \Vdash \langle\hat{\alpha}_{\langle\rangle_\mathcal{M}}\rangle\mathfrak{R}_\mathcal{M}(\psi)$. Namely $\ulcorner\mathcal{M}\urcorner, s \Vdash \mathfrak{R}_\mathcal{M}(\phi)$.

$\Leftarrow$) Suppose $\ulcorner\mathcal{M}\urcorner, s \Vdash \langle\hat{\alpha}_{\langle\rangle_\mathcal{M}}\rangle\mathfrak{R}_\mathcal{M}(\psi)$, then there exits a path $s \xrightarrow{e_{\sigma_1}} \cdots \xrightarrow{e_{\sigma_n}} t$ in $\ulcorner\mathcal{M}\urcorner$ such that $e_{\sigma_1} \cdots e_{\sigma_n} \in \mathcal{L}(\hat{\alpha}_{\langle\rangle_\mathcal{M}})$ with $\{\sigma_1, \ldots, \sigma_n\} \subseteq \langle\rangle_\mathcal{M}$. It follows that $exp_{Act}(e_{\sigma_1} \cdots e_{\sigma_n}) \subseteq exp_{Act}(\hat{\alpha}_{\langle\rangle_\mathcal{M}})$, since $\hat{\alpha}_{\langle\rangle_\mathcal{M}}$ is the maximal rewriting. Namely $s \xrightarrow{\sigma_1} \cdots \xrightarrow{\sigma_n} t$ in $\mathcal{M}$ and $\mathcal{L}(\sigma_1 \cdots \sigma_n) \subseteq \mathcal{L}(\alpha)$. By induction hypothesis, $\mathcal{M}, t \models \psi$, and thus $\mathcal{M}, s \models \phi$. □

Theorem 1 allows us to use the standard PDL model checking algorithm (e.g. [11]) to solve the problem over ALTS in a straightforward manner. We present an example here. Let us consider the ALTS $\mathcal{M}$ depicted in Fig. 1. Suppose we need to check whether the formula $\phi = \langle a \cdot (b \cdot a + c^*)\rangle p$ holds at state s. Then we first collect the set $\langle\rangle_\mathcal{M} = \{a, a \cdot c^* \cdot b, c\}$; then we compute the maximal rewriting of $a \cdot (b \cdot a + c^*)$ w.r.t $\langle\rangle_\mathcal{M}$, following the algorithm in [3]. It follows easily that $\mathfrak{R}_\mathcal{M}(\phi) = \langle e^*_{a \cdot c^* \cdot b} \cdot e_a \cdot e^*_c\rangle p$. According to Theorem 1, we only need to check whether $\ulcorner\mathcal{M}\urcorner, s \Vdash \mathfrak{R}_\mathcal{M}(\phi)$, where $\ulcorner\mathcal{M}\urcorner$ is the same graph as in Fig. 1 except that the labels become $e_{a \cdot c^* \cdot b}, e_a, e_c$ in an obvious way. A standard PDL model checking algorithm will return TRUE and thus we can conclude that $\mathcal{M}, s \models \phi$.

3.2 Complexity Analysis

Upper Bounds. We have shown that model checking PDL on ALTS can be reduced to model checking PDL on LTS plus the rewriting part. So the problem is in P time with an EXPSPACE-bounded oracle. So the complexity is $\text{P}^{\text{EXPSPACE}}$, which is EXPSPACE.

One might think the complexity is a bit scaring for practice. However, Lichtenstein and Pnueli argued that when analyzing the complexity of model checking, a distinction should be made between complexity in the size of the input structure and complexity in the size of the input formula. And it is often the complexity in size of the structure that is typically the computational bottleneck [12]. In a nutshell, *program complexity* refers to the complexity of the problem in terms of the size of the input module, assuming the formula is *fixed*. Clearly, in our case, the program complexity turns out to be LOGSPACE. This is important for practice since people might argue that the complexity of our algorithm is too high to be practical. However, in practice, usually the logic formula is small and in this case the algorithm still performs very well.

Lower Bound. We show that the upper bound established in previous section is essentially optimal. We shall exploit the regular expression rewriting problem (see Section 2.3) to prove the EXPSPACE lower bound of the problem of model checking ALTS w.r.t. a PDL formula. First, we have:

Theorem 2 ([3]) The problem of verifying the existence of a nonempty rewriting of a regular expression β w.r.t. a set $\mathcal{E}$ of regular expressions is EXPSPACE-complete.

We present a reduction as follows:

Lemma 1 Given a set of non-empty regular expressions $\mathcal{E} = \{\alpha_1, \cdots, \alpha_k\}$ and a regular expression β, there exists a pointed ALTS model $\mathcal{M}_\mathcal{E}, s$ and a PDL formula φ such that:

$\mathcal{M}_\mathcal{E}, s \models \varphi \iff$ there is a non-empty rewriting of β w.r.t. $\mathcal{E}$.

Proof: Given $\mathcal{E} = \{\alpha_1, \cdots, \alpha_k\}$ and β, we define the ALTS $\mathcal{M}_\mathcal{E}$ as $(\{s\}, \mathcal{E}, \rightarrow, V)$ where $\rightarrow = \{(s, e, s) \mid e \in \mathcal{E}\}$, V is an arbitrary valuation. Let $\varphi = \langle\beta\rangle\top$.

$\Rightarrow$) Suppose $\mathcal{M}_\mathcal{E}, s \models \langle\beta\rangle\top$. According to the definition, there is a path in $\mathcal{M}_\mathcal{E}$ with $s \xrightarrow{e'_1} s \cdots \xrightarrow{e'_m} s$ where $\{e'_1, \cdots, e'_m\} \subseteq \mathcal{E}$ and $\mathcal{L}(e'_1 \cdots e'_m) \subseteq \mathcal{L}(\beta)$. It follows that $e'_1 \cdots e'_m$ is a non-empty rewriting of β w.r.t. $\mathcal{E}$.

$\Leftarrow$) Suppose there is a nonempty rewriting β' of β w.r.t. $\mathcal{E}$. Since β is non-empty, there is a possibly empty word $e'_1 \cdots e'_m \in \mathcal{L}(\beta')$ where for each $1 \leq i \leq m$, $e'_i \in \mathcal{E}$. It is easy to see that $exp_\Sigma(e'_1 \cdots e'_m) \subseteq exp_\Sigma(\mathcal{L}(\beta'))$. Furthermore, according to the definition of the rewriting, $exp_\Sigma(\mathcal{L}(\beta')) \subseteq \mathcal{L}(\beta)$ and thus $exp_\Sigma(e'_1 \cdots e'_m) \subseteq \mathcal{L}(\beta)$. Since there exists a path in $\mathcal{M}_\mathcal{E}$ with $s \xrightarrow{e'_1} s \cdots \xrightarrow{e'_m} s$, $\mathcal{M}_\mathcal{E}, s \models \langle\beta\rangle\top$. This completes the proof. □

Theorem 1, Theorem 2 and Lemma 1 yield the main result of current section, as follows:

Theorem 3 The problem of model checking a PDL formula w.r.t. an ALTS is EXPSPACE-complete.

4 Axiomatization

In this section, we give a logical characterization of our semantics. Although the syntax of PDL does not change, the interpretation over ALTS results in a new semantics which differs from standard PDL considerably. For instance, the following axioms are valid in standard PDL. However, most of them do *not* hold anymore (in the right column, $\leftarrow$, if appears, denotes that the $\leftrightarrow$ connective should be replaced by $\leftarrow$ to keep the formula valid[2]).

Axioms	In our semantics
$[\alpha](\phi \to \psi) \to ([\alpha]\phi \to [\alpha]\psi)$	valid
$\langle\alpha_1 \cdot \alpha_2\rangle\phi \leftrightarrow \langle\alpha_1\rangle\langle\alpha_2\rangle\phi$	$\leftarrow$
$\langle\alpha_1 + \alpha_2\rangle\phi \leftrightarrow \langle\alpha_1\rangle\phi \vee \langle\alpha_2\rangle\phi$	$\leftarrow$
$\langle\alpha^*\rangle\phi \leftrightarrow (\phi \vee \langle\alpha\rangle\langle\alpha^*\rangle\phi)$	$\leftarrow$
$[\alpha^*](\phi \to [\alpha]\phi) \to (\phi \to [\alpha^*]\phi)$	invalid

In view of this, instead of the standard PDL axioms we propose the following new conditional axiomatization.

Definition 5 A deductive system AS

TAUTOLOGY	all the tautologies
K	$[\alpha](p \to q) \to ([\alpha]p \to [\alpha]q)$
SEQ	$[\alpha_1 \cdot \alpha_2]p \to [\alpha_1][\alpha_2]p$
$\star$	$[\alpha^*]p \to p$
Rules	
$\Box$	$\dfrac{p}{[\alpha]p}$
SUB	$\dfrac{\phi(p)}{\phi(\psi)}$
MP	$\dfrac{\phi, \phi \to \psi}{\psi}$
INCL	$\dfrac{\vdash_{\mathtt{KA}} \alpha + \alpha' = \alpha'}{[\alpha']p \to [\alpha]p}$

where KA is a complete Kleene algebra, for example as in [10], in acting as an oracle.

The rest of this section is devoted to showing that AS is sound and complete w.r.t to the class of all ALTS frames. First let us consider a special class of ALTS frames on which we can use an equivalent simple semantics for technical convenience. An ALTS frame is called *normal* if it satisfies the following properties:

- sequentiality: For any $\sigma, \sigma' \in Act^*$: $\xrightarrow{\sigma} \cdot \xrightarrow{\sigma'} \subseteq \xrightarrow{\sigma\cdot\sigma'}$;
- *-reflexivity: For any $\sigma \in Act^*$: if $\{\epsilon\} \in \mathcal{L}(\sigma)$ then $s \xrightarrow{\sigma} s$ for any $s \in S$;
- regularity: For any $\sigma, \sigma' \in Act^*$: $\mathcal{L}(\sigma) \subseteq \mathcal{L}(\sigma')$ implies that $\xrightarrow{\sigma} \subseteq \xrightarrow{\sigma'}$.

[2]Note that, the last induction axiom of PDL is not valid anymore, it makes the completeness proof easier than usual PDL.

Models based on the normal ALTS frames are called *normal* ALTS models. Now we can define an equivalent semantics $\models_0$ on the normal ALTS models as follows:

- For boolean cases: as before;
- For modal case:
  $\mathcal{M}, s \models_0 \langle\beta\rangle\phi \iff \exists t : s \xrightarrow{\beta} t$ and $t \models_0 \phi$.

We can saturate an arbitrary ALTS frame of PDL_Σ: $\mathcal{F} = (S, Act, \to)$ into a normal frame $R(\mathcal{F}) = (S, Act, \to_r)$ by adding transitions:

$$s \xrightarrow{\beta}_r t \iff \exists s \xrightarrow{\sigma_1} s_1 \xrightarrow{\sigma_1} \cdots \xrightarrow{\sigma_n} s_n \text{ and } \mathcal{L}(\sigma_1\sigma_2\ldots\sigma_n) \subseteq \mathcal{L}(\beta)$$

$R(\mathcal{M})$ is the saturated model which keeps the valuation the same but saturates the frame of $\mathcal{M}$. It is easy to see that $\models_0$ coincides with $\models$ on normal models:

Proposition 1 Given an ALTS $\mathcal{M} = (S, Act, \to, V)$, for any PDL formula ϕ:

$$\mathcal{M}, s \models \phi \iff R(\mathcal{M}), s \models_0 \phi \iff R(\mathcal{M}), s \models \phi$$

Since all the normal ALTS frames are ALTS frames and all the ALTS frames can be saturated into normal ALTS frames, it follows from the above proposition that $\Delta \models \phi \iff \Delta \models_0 \phi$, where Δ is a set of PDL formulas.

It is easy to check the following lemma:

Lemma 2 For any normal ALTS frame $\mathcal{F}$ and any two regular expressions σ and σ', if $\models_{\mathtt{KA}} \sigma + \sigma' = \sigma'$ then $\models_0 [\sigma']p \to [\sigma]p$.

Lemma 3 For any ALTS frame $\mathcal{F}$: $\mathcal{F}$ satisfies sequentiality $\iff \mathcal{F} \models_0$ SEQ.

Lemma 4 For any ALTS frame $\mathcal{F}$: $\mathcal{F}$ satisfies *-reflexivity implies $\mathcal{F} \models_0 \star$.

From above lemma, and the completeness of Kleene Algebra [10], it is straightforward to establish:

Theorem 4 [Soundness] AS is sound for normal ALTS frames.

Note that the $\star$ axiom does not correspond to $*$-reflexivity by itself, but in presence of the other two properties[3]:

Lemma 5 If an ALTS frame $\mathcal{F}$ satisfies regularity, sequentiality and $\mathcal{F} \models_0 \star$ then $\mathcal{F}$ is normal.

[3]That is why we don't include a rule like: $\dfrac{[\alpha]\phi}{\phi}$ if $\epsilon \in \mathcal{L}(\alpha)$.

Proof: Suppose $\mathcal{F}$ satisfies regularity and sequentiality, we only need to show $\mathcal{F}$ satisfies *-reflexivity: for any regular expression σ appearing in PDL, if $\epsilon \in \mathcal{L}(\sigma)$ then $\xrightarrow{\sigma}$ is reflexive. We prove this by induction on the structure of σ.

- If $\sigma = \sigma'^*$ then it is straightforward to check that $\xrightarrow{\sigma}$ is reflexive since $\mathcal{F} \vDash_0 *$.
- If $\sigma = \sigma_1 + \sigma_2$ then $\epsilon \in \mathcal{L}(\sigma_1)$ or $\epsilon \in \mathcal{L}(\sigma_2)$. By induction hypothesis $\xrightarrow{\sigma_1}$ is reflexive or $\xrightarrow{\sigma_2}$ is reflexive. From regularity, $\xrightarrow{\sigma_1} \subseteq \xrightarrow{\sigma}$ and $\xrightarrow{\sigma_2} \subseteq \xrightarrow{\sigma}$. So $\xrightarrow{\sigma}$ is reflexive.
- If $\sigma = \alpha \cdot \sigma_2$ then $\epsilon \in \mathcal{L}(\sigma_1)$ and $\epsilon \in \mathcal{L}(\sigma_2)$. By induction hypothesis $\xrightarrow{\sigma_1}$ and $\xrightarrow{\sigma_2}$ are reflexive. From sequentiality, $\xrightarrow{\sigma_1 \cdot \sigma_2} \subseteq \xrightarrow{\sigma}$. So $\xrightarrow{\sigma}$ is reflexive. □

Completeness follows from the standard canonical model construction.

Theorem 5 [Completeness] For any set of PDL formulas $\Delta \cup \{\phi\}$: $\Delta \vDash_0 \phi \implies \Delta \vdash_{\mathsf{AS}} \phi$. Namely AS is strongly complete for normal ALTS frames w.r.t $\vDash_0$. Thus AS is strongly complete for all ALTS frames.

Proof: Note that AS induces a normal logic[4]. Therefore it is strongly complete with respect to its canonical model $\mathcal{M}^c = (S^c, \Sigma^*, \longrightarrow^c, V^c)$ according to canonical model theorem[5]. We only need to show that the canonical model $\mathcal{M}^c$ is indeed a model based on normal ALTS frame. Since S^c is the set of AS-maximal consistent sets, $\mathcal{M}^c \vDash_0 * \wedge SEQ$. From Lemma 3 and 5, we only need to show the canonical model satisfies regularity:

$$\text{For any } \sigma, \sigma' \in \Sigma^*,$$
$$\mathcal{L}(\sigma) \subseteq \mathcal{L}(\sigma') \text{ implies } \xrightarrow{\sigma}{}^c \subseteq \xrightarrow{\sigma'}{}^c .$$

Suppose there are regular expressions σ, σ' such that $\mathcal{L}(\sigma) \subseteq \mathcal{L}(\sigma')$ and $\exists s, t : s \xrightarrow{\sigma}{}^c t$ in the canonical model. From the definition of $\xrightarrow{\sigma}{}^c$, we have for all $\psi : \psi \in t \Rightarrow \langle\sigma\rangle\psi \in s$. Since s is a maximal consistent set, then from INCL we have for all $\psi : \langle\sigma\rangle\psi \rightarrow \langle\sigma'\rangle\psi \in s$. Therefore by applying MP rule we have for all $\psi \in t : \langle\sigma'\rangle\psi \in s$. It follows, by definition, that $s \xrightarrow{\sigma'}{}^c t$. □

Strong completeness implies the compactness:

Corollary 1 [Compactness] PDL w.r.t ALTS is model compact. Namely if all the finite subsets of Γ are satisfiable then Γ is satisfiable.

[4] A logic theory is *normal* if it contains all the instances of tautologies, K axiom and closed under MP, SUB and □.

[5] S^c is the set of all AS−maximal consistent sets, $w \xrightarrow{\sigma} v$ if for all ψ, $\psi \in v \Rightarrow \langle\sigma\rangle\psi \in w$, $V^c = \{s \in S^c \mid p \in s\}$. Readers are referred to the textbook [2] for more details about canonical model theorem.

Remark 1 Recall that standard PDL is not model compact: considering the set $\Gamma = \{\langle a^*\rangle p, \neg p, \neg\langle a\rangle p, \neg\langle a\rangle\langle a\rangle p, \cdots\}$, any finite subset of Γ is satisfiable, yet not the whole Γ. However, Γ is satisfiable on a single pointed ALTS model with a single reflexive a^*-transition.

5 Satisfiability

In this section, we turn to the satisfiability checking problem. The basic idea is to reduce this problem to traditional PDL satisfiability checking. However, clearly this can not be done in a straightforward way, since their semantics do not coincide, as observed in previous section.

For technical reasons, let us consider the equivalent positive PDL$^+$ language

$$\varphi \quad ::= \quad \top \mid \bot \mid p \mid \overline{p} \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid [\alpha]\varphi \mid \langle\alpha\rangle\varphi$$

where p and $\overline{p}$ (negation of p) are in a set lit of literals of basic propositions and α ranges over programs as in PDL. It is a standard exercise to transform a PDL formula to an equivalent PDL$^+$ formula and vice versa.

Given a PDL$^+$ formula ϕ, let $\langle\rangle_\phi$ be the set $\{\alpha \mid \alpha \text{ appears in } \phi \text{ in form of } \langle\alpha\rangle\}$. We now prove that if a formula is satisfiable then it is satisfiable in a certain class of models.

Proposition 2 Given a PDL$^+$ formula ϕ, ϕ is satisfiable $\iff$ ϕ is satisfiable in a model that only contains α-transitions for $\alpha \in \langle\rangle_\phi$.

Proof: $\Leftarrow$ is straightforward. We now prove $\Rightarrow$:
Suppose ϕ is satisfiable then there is an ALTS model $\mathcal{M} = (S, Act, \rightarrow, V)$ such that $\exists s \in S : \mathcal{M}, s \vDash \phi$. From proposition 1, $R(\mathcal{M}), s \vDash \phi$. Based on $R(\mathcal{M})$ we build the model $\mathcal{M}' = \{S, Act', \rightarrow', V\}$ where:

$$Act' = \langle\rangle_\phi \text{ and } s \xrightarrow{\alpha}{}' t \text{ in } \mathcal{M}' \iff s \xrightarrow{\alpha}{}_r t \text{ in } R(\mathcal{M}).$$

Namely we cut off all the transitions in $R(\mathcal{M})$ but the ones labelled by some $\alpha \in \langle\rangle_\phi$. We claim: $\mathcal{M}', s \vDash \phi$. We do induction on the structure of ϕ :

- For atomic and boolean cases, trivial.
- $\phi = \langle\alpha\rangle\psi$: since $R(\mathcal{M}), s \vDash \phi$ then $\exists t \in S$ such that $R(\mathcal{M}), t \vDash \psi$ and $s \xrightarrow{\alpha}{}_r t$. By definition $s \xrightarrow{\alpha}{}' t$. By induction hypothesis, $\mathcal{M}', t \vDash \psi$ thus $\mathcal{M}', s \vDash \phi$.
- $\phi = [\beta]\psi$: since $R(\mathcal{M}), s \vDash \phi$ then for all t such that $s \xrightarrow{\beta}{}_r t$, $R(\mathcal{M}), t \vDash \psi$. By induction hypothesis, $\mathcal{M}', t \vDash \psi$. Note that if there exists t such that $s \xrightarrow{\alpha'_1}{}' \cdots \xrightarrow{\alpha'_n}{}' t$ in $\mathcal{M}'$, and $\mathcal{L}(\alpha'_1 \cdot \cdots \cdot \alpha'_n) \subseteq \mathcal{L}(\beta)$ then $s \xrightarrow{\beta}{}_r t$ in $R(\mathcal{M})$. Therefore for all β−reachable states t in S, $\mathcal{M}', t \vDash \psi$. It means $\mathcal{M}', s \vDash \phi$. □

Given a PDL$^+$ formula ϕ, we define a rewriting of ϕ, which substitutes every instance of β in $[\beta]\psi$ by its maximal $\Sigma_{\langle\rangle_\phi}$-rewriting $\hat{\beta}_{\langle\rangle_\phi}$. Recall that, according to *regular rewriting*, $\hat{\beta}_{\langle\rangle_\phi}$ is regular expression over the alphabet $\Sigma_{\langle\rangle_\phi} = \{e_\alpha \mid \alpha \in \langle\rangle_\phi\}$ where each e_α is a new action name.

Definition 6 [Rewriting] Given a PDL$^+$ formula ϕ, $\mathfrak{R}(\phi)$ is the rewriting of ϕ in language $\text{PDL}^+_{\Sigma_{\langle\rangle_\phi}}$ defined by:

- $\mathfrak{R}(p) = p$ where $p \in lit \cup \{\top, \bot\}$.
- $\mathfrak{R}(\psi_1 \wedge \psi_2) = \mathfrak{R}(\psi_1) \wedge \mathfrak{R}(\psi_2)$.
- $\mathfrak{R}(\psi_1 \vee \psi_2) = \mathfrak{R}(\psi_1) \vee \mathfrak{R}(\psi_2)$.
- $\mathfrak{R}(\langle\alpha\rangle(\psi)) = \langle e_\alpha\rangle\mathfrak{R}(\psi)$.
- $\mathfrak{R}([\beta]\psi) = [\hat{\beta}_{\langle\rangle_\phi}]\mathfrak{R}(\psi)$.

Proposition 3 Given a PDL$^+$ formula ϕ, ϕ is satisfiable $\iff$ $\mathfrak{R}(\phi)$ is satisfiable w.r.t. standard PDL semantics.

Proof: $\Rightarrow$) Suppose ϕ is satisfiable, then from proposition 2, we know that ϕ is satisfiable in an ALTS model $\mathcal{M}$ that only contains α-transitions for $\alpha \in \langle\rangle_\phi$. Note that we can also treat $\mathcal{M}$ as an LTS over the action set $\Sigma_{\langle\rangle_\phi}$, which we denote by $\mathcal{G}$. Namely, $\mathcal{G}$ is the same as $\mathcal{M}$ except that the transition is renamed. We now show $\mathcal{G}, s \Vdash \mathfrak{R}(\phi)$ by induction on the structures of $\mathfrak{R}(\phi)$:

- For atomic and boolean cases, trivial.
- $\mathfrak{R}(\phi) = \langle e_\alpha\rangle\mathfrak{R}(\psi)$, where $\alpha \in \langle\rangle_\phi$. Since $\mathcal{M}, s \vDash \phi$, there exists some $s \xrightarrow{\alpha} s'$ with $\mathcal{M}, s' \models \psi$. According to our construction, in $\mathcal{G}$, $s \xrightarrow{e_\alpha} s'$. By induction hypothesis, $\mathcal{G}, s' \Vdash \mathfrak{R}(\psi)$ in $\mathcal{G}$. It follows from the semantics of traditional PDL that $\mathcal{G}, s \Vdash \mathfrak{R}(\phi)$.
- For $\mathfrak{R}(\phi) = [\hat{\beta}_{\langle\rangle_\phi}]\mathfrak{R}(\psi)$: Since $\mathcal{M}, s \models \phi$, for any sequence of transitions $s \xrightarrow{\alpha_1} \cdots \xrightarrow{\alpha_n} s'$ with $n \geq 0$, $\mathcal{L}(\alpha_1 \cdots \alpha_n) \subseteq \mathcal{L}(\beta)$ implies $\mathcal{M}, s' \models \psi$. Now let us consider any sequence of transitions $s \xrightarrow{e_{\alpha'_1}} \cdots \xrightarrow{e_{\alpha'_m}} s_m$ with $e_{\alpha'_1} \cdots e_{\alpha'_m} \in \mathcal{L}(\hat{\beta}_{\langle\rangle_\phi})$ in $\mathcal{G}$. Note that $e_{\alpha'_1}, \cdots, e_{\alpha'_m}$ are single actions, hence $\mathcal{L}(e_{\alpha'_1} \cdots e_{\alpha'_m}) \subseteq \mathcal{L}(\hat{\beta}_{\langle\rangle_\phi})$. Since $\hat{\beta}_{\langle\rangle_\phi}$ is a $\langle\rangle_\phi-$rewriting of β, $\mathcal{L}(\alpha'_1 \cdots \alpha'_m) \subseteq \mathcal{L}(\beta)$. Therefore $\mathcal{M}, s_m \models \psi$. By induction hypothesis, $\mathcal{G}, s_m \Vdash \mathfrak{R}(\psi)$. It follows that $\mathcal{G}, s \Vdash \phi$.

$\Leftarrow$) Suppose $\mathfrak{R}(\phi)$ is satisfiable, there is a pointed LTS $\mathcal{G}, s$ over action set $\Sigma_{\langle\rangle_\phi}$ such that $\mathcal{G}, s \Vdash \mathfrak{R}(\phi)$. Clearly, we can construct a corresponding ALTS $\mathcal{M}$ which is the same as $\mathcal{G}$ except that for any transition $e_\alpha \in \Sigma_{\langle\rangle_\phi}$ in $\mathcal{G}$, we take the transition $\alpha \in \langle\rangle_\phi$ in $\mathcal{M}$. We now show $\mathcal{M}, s \vDash \phi$ by induction on the structures of ϕ:

- For atomic and boolean cases, trivial.
- $\phi = \langle\alpha\rangle\psi$, where $\alpha \in \langle\rangle_\phi$. Since $\mathcal{G}, s \Vdash \mathfrak{R}(\phi)$, namely $\mathcal{G}, s \Vdash \langle e_\alpha\rangle\mathfrak{R}(\psi)$, there exists some $s \xrightarrow{e_\alpha} s'$ in $\mathcal{G}$ with $s' \Vdash \mathfrak{R}(\psi)$. According to our construction, $s \xrightarrow{\alpha} s'$ in $\mathcal{M}$. By induction hypothesis, $\mathcal{M}, s' \models \psi$. It follows from our semantics that $\mathcal{M}, s \vDash \phi$.
- $\phi = [\beta]\psi$: Since $\mathcal{G}, s \Vdash \mathfrak{R}(\phi)$, namely $\mathcal{G}, s \Vdash [\hat{\beta}_{\langle\rangle_\phi}]\mathfrak{R}(\psi)$, $s \xrightarrow{e_{\alpha_1}} \cdots \xrightarrow{e_{\alpha_m}} s'$ and $e_{\alpha_1} \cdots e_{\alpha_m} \in \mathcal{L}(\hat{\beta}_{\langle\rangle_\phi})$ implies $\mathcal{G}, s' \vDash \mathfrak{R}(\psi)$. Take arbitrary t such that $s \xrightarrow{\alpha'_1} \cdots \xrightarrow{\alpha'_n} t$ in $\mathcal{M}$ and $\mathcal{L}(\alpha'_1 \cdots \alpha'_n) \subseteq \mathcal{L}(\beta)$. Since $\hat{\beta}_{\langle\rangle_\phi}$ is the maximal $\Sigma_{\langle\rangle_\phi}-$rewriting of β, $\mathcal{L}(e_{\alpha'_1} \cdots e_{\alpha'_n}) \subseteq \mathcal{L}(\hat{\beta}_{\langle\rangle_\phi})$. Since $e_{\alpha'_1}, \ldots, e_{\alpha'_n}$ are atomic, $e_{\alpha'_1} \cdots e_{\alpha'_n} \in \mathcal{L}(\hat{\beta}_{\langle\rangle_\phi})$. Hence $\mathcal{G}, t \Vdash \mathfrak{R}(\psi)$. By induction hypothesis, $\mathcal{M}, t \vDash \psi$. Therefore $\mathcal{M}, s \vDash \phi$. □

Remark 2 This result is somewhat surprising. Note that our semantics and traditional PDL semantics differs as shown in the previous section. However, they coincide after the rewriting. For example, $\phi = \langle a \cdot b\rangle p \wedge [a][b]\neg p$ is satisfiable w.r.t our semantics, but not in traditional PDL while $\mathfrak{R}(\phi) = \langle e_{a\cdot b}\rangle p \wedge [\delta][\delta]\neg p$ is satisfiable in traditional PDL semantics.

From proposition 3, we managed to reduce satisfiability checking of PDL over ALTS to traditional PDL satisfiability checking, which has been extensively studied in literature, see e.g. [8], and is EXPTIME-complete. Note the regular expression rewriting can be done in EXPSPACE. These entail that the satisfiability checking of PDL over ALTS can be done in EXPSPACE. Now we prove the lower bound by reducing regular expression rewriting problem to the satisfiability problem.

Lemma 6 Given a set of regular expressions $\mathcal{E} = \{\alpha_1, \ldots, \alpha_k\}$, another regular expression β, which are over an alphabet Σ, there exists a PDL$-$formula $\phi_{\mathcal{E},\beta}$ such that $\phi_{\mathcal{E},\beta}$ is satisfiable $\iff$ there does not exist a non-empty rewriting of β w.r.t $\mathcal{E}$.

Proof: (Sketch) Given $\mathcal{E} = \{\alpha_1, \ldots, \alpha_k\}$ and β, let

$$\phi_{\mathcal{E},\beta} = [\beta]p \wedge [(\alpha_1 + \cdots + \alpha_k)^*](\neg p \wedge \langle\alpha_1\rangle\neg p \wedge \cdots \wedge \langle\alpha_k\rangle\neg p)$$

$\Rightarrow$) Suppose $\phi_{\mathcal{E},\beta}$ is satisfiable, then according to proposition 2 there is a pointed ALTS $\mathcal{M}, s_0$ containing only $\alpha_1, \cdots, \alpha_n \in \mathcal{E}$ transitions such that $\mathcal{M}, s \vDash \phi_{\mathcal{E},\beta}$. Since $\mathcal{M}, s_0 \vDash [(\alpha_1 + \cdots + \alpha_k)^*](\neg p \wedge \langle\alpha_1\rangle\neg p \wedge \cdots \wedge \langle\alpha_k\rangle\neg p)$, $s_0 \models \neg p \wedge \langle\alpha_1\rangle\neg p \wedge \cdots \wedge \langle\alpha_k\rangle\neg p$, and thus from s_0, for each α_i, there must be some $s_0 \xrightarrow{\alpha_i} s_i$ and according to our semantics, $s_i \vDash \neg p \wedge \langle\alpha_1\rangle \wedge \cdots \wedge \langle\alpha_k\rangle\neg p$. Repeat this process, it is not difficult to see that for each

$\sigma_0 \cdots \sigma_m \in \mathcal{L}((\alpha_1 + \cdots + \alpha_n)^*)$ where $\sigma_i \in \mathcal{E}$, $\mathcal{M}$ contains a sequence of transitions $s_0 \xrightarrow{\sigma_0} s_1 \xrightarrow{\sigma_1} \cdots \xrightarrow{\sigma_m} s_m$ and for any $i \leq m$, $\mathcal{M}, s_i \vDash \neg p$.

Moreover, since $s_0 \models [\beta]p \wedge [(\alpha_1 + \cdots + \alpha_k)^*](\neg p \wedge \langle\alpha_1\rangle\neg p \wedge \cdots \wedge \langle\alpha_k\rangle\neg p)$, $s_0 \models [\beta]p \wedge \neg p$ and thus $\varepsilon \notin \mathcal{L}(\beta)$. Hence ε can not be a rewriting of β. Furthermore, since $\mathcal{M}, s \vDash [\beta]p$, it is easy to see that for any sequence $e \in \mathcal{L}((e_{\alpha_1} + \cdots + e_{\alpha_k})^*)$, $exp_\Sigma(\mathcal{L}(e)) \not\subseteq \mathcal{L}(\beta)$, because otherwise, we can find a path in $\mathcal{M}$ such that this path leads to a state where p holds, which is a contradiction. Hence there is no non-empty rewriting of β w.r.t. $\mathcal{E}$.

$\Leftarrow$) Suppose there is no non-empty rewriting of β w.r.t. $\mathcal{E}$ then for all $e \in \mathcal{L}((e_{\alpha_1} + \cdots + e_{\alpha_k})^*) : exp_\Sigma(\mathcal{L}(e)) \not\subseteq \mathcal{L}(\beta)$. We can build a model $\mathcal{M} = \{\{s\}, \mathcal{E}, \rightarrow, V\}$ where $\rightarrow = \{(s, \alpha, s) \mid \alpha \in \mathcal{E}\}$ and $V(s) = \{\neg p\}$. It is clear that $\mathcal{M}, s \vDash \phi$. □

From the above lemma and theorem 2 we have:

Theorem 6 The satisfiability problem of PDL w.r.t. ALTS is EXPSPACE-complete.

6 Conclusion and Future works

We have performed a thorough study of Propositional Dynamic Logic over accelerated labelled transition systems. We mainly investigated three problems: model checking, axiomatization and satisfiability checking. We show that the model checking problem of this logic is EXPSPACE-complete while the program complexity turns out to be NLOGSPACE-complete. This answers an open question in [16]; We also provide a sound and complete axiomatization for PDL which involves Kleene Algebra as an Oracle; Furthermore, we solve the satisfiability decision problem by a reduction to the satisfiability of PDL in the traditional semantics (w.r.t. LTS). The complexity is EXPSPACE-complete as well.

There are a lot of avenues for future study. First, there are a number of extensions of PDL (e.g. the test operator) and we are interested in what will happen if they meet ALTS. Furthermore, in order to apply ALTS to abstract model checking of liveness properties, as sketched in [16], some open problems remain, for instance, how can an abstraction with accelerated transitions be computed automatically? [16] hints at the relation with automated termination provers. Our study shows that the model checking problem with accelerated transitions is hard. So another interesting question is how to add the minimal number of accelerated transitions, in order to prove a certain liveness property.

Acknowledgement. The first author is partially supported by the Dutch Bsik project BRICKS, the Chinese national 863 program (2007AA01Z178), NSFC (60736015) and JSNSF (BK2006712); The third author is partially supported by the Dutch NWO project VEMPS (612.000.528).

References

[1] G. Bruns and P. Godefroid. Model checking partial state spaces with 3-valued temporal logics. In *Proc. CAV'99*, LNCS 1633, pp. 274-287, Springer, 1999.

[2] P. Blackburn, M. de Rijke and Y. Venema. Modal logic. Cambridge University Press, 2002.

[3] D. Calvanese, G. De Giacomo, M. Lenzerini and M. Vardi. Rewriting of regular expressions and regular path queries. *Journal of Computer and System Sciences*, 64(3): 443-465, 2002.

[4] E. Clarke, Orna Grumberg and D. Peled. Model Checking, MIT Press, 2000.

[5] M. Fischer and R. Ladner. Propositional dynamic logic of regular programs, *Journal of Computer and System Sciences*, 18(2):194-211, 1979.

[6] P. Godefroid, M. Huth and R. Jagadeesan. Abstraction-based model checking using modal transition systems. In *Proc. of CONCUR'01*, LNCS 2154, pp. 426-440. Springer, 2001.

[7] D. Giammarresi and R. Montalbano. Deterministic generalized automata. *Theoremtical Computer Science*, 215: 191-208, 1999.

[8] D. Harel, D. Kozen and J. Tiuryn. Dynamic Logic, MIT Press, Cambridge, MA, 2000.

[9] Y. Han and D. Wood. The generalization of generalized automata: expression automata. *International Journal of Foundations of Computer Science*, 16(3): 499-510, 2005.

[10] D. Kozen. A completeness theorem for Kleene algebras and the algebra of regular events. *Jounral of Information and Computation*, 110(2):366-390, 1994.

[11] M. Lange. Model checking propositional dynamic logic with all extras. *Journal of Applied Logic*, 4:39-49, 2006.

[12] O. Lichtenstein, and A. Pnueli. Checking that finite state concurrent programs satisfy their linear specification. In *Proc. POPL'85*, pp. 97-107, ACM Press, 1985.

[13] M. Leucker, C. Sánchez. Regular linear temporal logic. In *Proc. ICTAC'07*, LNCS 4711, pp. 291-305, Springer, 2007.

[14] K. Larsen and B. Thomsen. A modal process logic. In *Proc. of LICS'88*, pp. 203-210, IEEE computer society, 1988.

[15] R. Mateescu and M. Sighireanu. Efficient on-the-fly model-checking for regular alternation-free mu-calculus. *Science of Computer Programming*, 46(3):255-281, 2003.

[16] M. Valero Espada and J. van de Pol. Accelerated modal abstractions of labelled transition systems. In *Proc. AMAST'06*, LNCS 4019, pp. 338-352, Springer, 2006.