Theoretical Computer Science 16 (1981) 321–328
North-Holland Publishing Company

NOTE

# PROPOSITIONAL DYNAMIC LOGIC IS WEAKER WITHOUT TESTS*

Francine BERMAN
*Department of Computer Sciences, Purdue University, West Lafayette, IN 47907, U.S.A.*

Michael PATERSON
*Department of Computer Science, University of Warwick, Coventry CV4 7AL, United Kingdom*

**Abstract.** We show that Test-free Propositional Dynamic Logic ($PDL_0$) is less expressive than Propositional Dynamic Logic, i.e. we show that there is a formula of PDL using the conditional program operator ? which is not equivalent to any formula of PDL using only the regular program operators concatenation, branching and iteration. Extensions of the problem are discussed in the conclusion.

Communicated by A. Meyer
Received December 1980

## 1. Introduction

Propositional Dynamic Logic (PDL) is a programming meta-language, i.e. a formal language for reasoning about structured programs. Using flowchart schemes as a model, PDL has program constructs for sequencing, nondeterministic branching, iteration and tests. The addition of the conditional operator to the set of regular operators enables us to express **while** and **if–then–else** statements. But does the conditional construct really increase expressive power?

In this paper, we show that PDL has more expressive power than $PDL_0$ (PDL without tests). Furthermore, we note in Section 5 that the ability to nest conditionals increases expressive power with each additional level of nesting.

To show that $PDL_0$ is less expressive than PDL, our strategy is to exhibit a class of models $\{\mathscr{A}_m\}_{m>1}$ in which no test-free formula is equivalent to a given formula $z$. The proof goes roughly as follows:

(1) First we define the formula $z$ and the class of cyclic models $\{\mathscr{A}_m\}$. Then assume towards a contradiction that $p$ is a formula in $PDL_0$ equivalent to $z$.

* This research was supported in part by the National Science Foundation through Grant Nos. DCR74-12997-A01, MCS77-02474 and MCS80-05387.

0304-3975/81/0000–0000/$02.75 

(2) We reduce $p$ to an equivalent formula $p'$ in a normal form to be described later.

(3) We show that for any $p'$ in normal form, any 'large enough' model $\mathscr{A}_m$ and for particular states $w_k$, $w_{k+m}$ in $\mathscr{A}_m$, $p'$ is satisfiable at $w_k$ in $\mathscr{A}_m$ iff $p'$ is satisfiable at $w_{k+m}$ in $\mathscr{A}_m$.

(4) This provides a contradiction since for similar $m$ and $k$, $z$ is satisfiable at $w_k$ in $\mathscr{A}_m$ but not at $w_{k+m}$ in $\mathscr{A}_m$.

In the conclusion, we discuss extensions of the problem.

## 2. Definitions and main result

We briefly define the syntax and semantics of PDL and its restriction to $\mathrm{PDL}_0$. For a more complete discussion see [3].

*Syntax of PDL*

The basic objects for PDL are two sets of primitives:

$\Phi_0$: the basic formulas (propositional variables),

$\Sigma_0$: the basic programs.

Programs and formulas are defined inductively:

(i) $\Theta$ (the null program) and basic programs are programs.

(ii) If $a$ and $b$ are programs and $p$ is a formula then $a;b$, $a\cup b$, $p?$ and $a^*$ are programs.

(iii) **True**, **false**, and basic formulas are formulas.

(iv) If $p$ and $q$ are formulas and $a$ is a program then $p\vee q$, $\neg p$ and $\langle a\rangle p$ are formulas.

We let $P, Q, R, \ldots$ represent members of $\Phi_0$ and $A, B, C, \ldots$ represent members $\Sigma_0$. The letters $p, q, r$ are reserved as metavariables for formulas, and $a, b, c, \ldots$ as metavariables for programs.

Note that the programs are called regular programs because of their similarity to regular expressions. Intuitively, we can make the correspondence:

| | |
|---|---|
| "$a;b$" | means "Begin $a;b$ end." |
| "$a\cup b$" | means "Nondeterministically do $a$ or $b$." |
| "$a^*$" | means "Repeat $a$ $n$ times where $n\geqslant 0$ is chosen non-deterministically." |
| "$p?$" | means "Test $p$ and proceed only if true." |
| "$\langle a\rangle p$" | means "After some execution of program $a$, formula $p$ is true." |

*Semantics of PDL*

A *structure* or *standard model* $\mathscr{A}$ of PDL is a triple $\mathscr{A}=(W^{\mathscr{A}}, \pi^{\mathscr{A}}, \rho^{\mathscr{A}})$ where $\pi^{\mathscr{A}}:\Phi_0\to 2^{W^{\mathscr{A}}}$ and $\rho^{\mathscr{A}}:\Sigma_0\to 2^{W^{\mathscr{A}}\times W^{\mathscr{A}}}$. We extend $\rho^{\mathscr{A}}$ to all programs and $\pi^{\mathscr{A}}$ to all formulas inductively as follows:

$\rho^{\mathscr{A}}(\Theta)=\emptyset$,
$\rho^{\mathscr{A}}(a;b)=\rho^{\mathscr{A}}(a)\circ\rho^{\mathscr{A}}(b)$ (composition of relations),
$\rho^{\mathscr{A}}(a\cup b)=\rho^{\mathscr{A}}(a)\cup\rho^{\mathscr{A}}(b)$ (union of relations),
$\rho^{\mathscr{A}}(a^*)=(\rho^{\mathscr{A}}(a))^*$ (the reflexive and transitive closure of a relation),
$\rho^{\mathscr{A}}(p?)=\{(w,w)\mid w\in\pi^{\mathscr{A}}(p)\}$,
$\pi^{\mathscr{A}}(\mathbf{true})=W^{\mathscr{A}}$,
$\pi^{\mathscr{A}}(\mathbf{false})=\emptyset$,
$\pi^{\mathscr{A}}(p\vee q)=\pi^{\mathscr{A}}(p)\cup\pi^{\mathscr{A}}(q)$,
$\pi^{\mathscr{A}}(\neg p)=W^{\mathscr{A}}-\pi^{\mathscr{A}}(p)$,
$\pi^{\mathscr{A}}(\langle a\rangle p)=\{w\in W^{\mathscr{A}}\mid \exists v((w,v)\in\rho^{\mathscr{A}}(a)$ and $v\in\pi^{\mathscr{A}}(p))\}$.

The superscripts may be dropped when the context is clear.

We write $\mathscr{A}, w\models p$ just in case $w\in\alpha^{\mathscr{A}}(p)$. We say that $p$ and $q$ are *equivalent in a structure* $\mathscr{A}$ (abbreviated $p\equiv q$ *in* $\mathscr{A}$) if for all $w\in W^{\mathscr{A}}$, $\mathscr{A}, w\models p$ iff $\mathscr{A}, w\models q$. In addition, $p$ and $q$ are *equivalent* if they are equivalent in all structures.

*$PDL_0$*

Let $\mathrm{PDL}_0$ be the restriction of PDL to test-free regular programs, i.e. programs which do not contain the symbol "?". Note that in $\mathrm{PDL}_0$, every program is representable by an expression over $\Sigma_0$.

## 3. PDL modulo a family of structures

Our main result establishes that PDL has more expressive power than $\mathrm{PDL}_0$.

**Main Theorem.** *Let $z$ be the PDL formula $\langle(P?;A)^*;\neg P?;A;P?\rangle$* **true**. *There is no formula in* $\mathrm{PDL}_0$ *equivalent to $z$.*

The full proof of this theorem will be given in Section 4, but we must first introduce the following family of structures:

For $m>1$, let

$$\mathscr{A}_m=(W^m,\pi^m,\rho^m)$$

where

$$W^m=\{w_0,\ldots,w_{2m}\},$$
$$\rho^m(A)=\{(w_i,w_{i+1})\mid i=0,\ldots,2m-1\}\cup\{(w_{2m},w_0)\},$$
$$\rho^m(B)=\emptyset\quad\text{for } B\in\Sigma_0-\{A\},$$
$$\pi^m(P)=W^m-\{w_{m-1},w_{2m-1},w_{2m}\},$$
$$\pi^m(Q)=W^m\quad\text{for } Q\in\Phi_0-\{P\}.$$

Observe that we have (see Fig. 1) embedded in the structure $\mathcal{A}_m$ (where $P$ is the propositional variable and $A$ is the basic program referred to in the main theorem). It is clear that $\mathcal{A}_m, w_0 \models z$ but $\mathcal{A}_m, w_m \not\models z$.

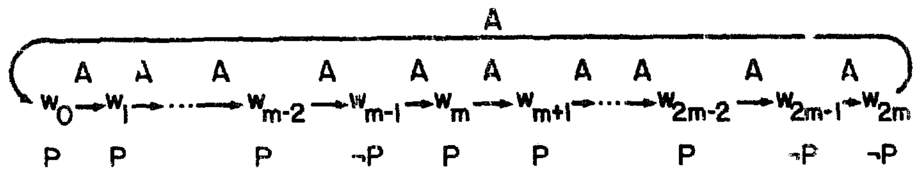


Fig. 1.

**Reduction Lemma 1.** *For any $p \in \mathrm{PDL}_0$ there is a $p' \in \mathrm{PDL}_0$ such that $p \equiv p'$ in $\mathcal{A}_m$ for all $m > 1$ and the only basic program occurring in $p'$ is $A$.*

**Proof.** Let $p'$ result from $p$ by replacing all occurrences of $B \subset \Sigma_0 - \{A\}$ by $\Theta$. Since $\rho^m(\Theta) = \rho^m(B)$, it follows that $\pi^m(p) = \pi^m(p')$. Further, the only basic program occurring in $p'$ is $A$.

**Reduction Lemma 2.** *For any $p \in \mathrm{PDL}_0$ in which the only basic program occurring is $A$, there is an equivalent $p' \in \mathrm{PDL}_0$ such that the only subformulas $\langle a \rangle q$ which occur in $p'$ have either $a = A$ or $a = (A^n)^*$ for some $n \geq 1$.*

**Proof.** By hypothesis, in any subformula $\langle a \rangle q$ of $p$, $a$ is a regular expression over $\{A\}$. Theorem 3.1.2 of [4] states that over a single symbol alphabet, the regular sets are precisely those which are ultimately periodic. It follows that any regular set over $\{A\}$ can be expressed in the form $b \cup c(A^n)^*$ for some $n \geq 1$ where $b, c$ are star-free (possibly empty) expressions. Note that the empty string can be represented by $\Theta^*$. For each hypothesized formula $p$, replace each subformula $\langle a \rangle q$ by the appropriate formula $\langle b \cup c;(A^n)^* > q$ and repeatedly use the following equivalences to eliminate $\cup$, ;, $\theta$:

(i) $\langle d \cup e \rangle q \equiv \langle d \rangle q \vee \langle e \rangle q$,
(ii) $\langle d;e \rangle q \equiv \langle d \rangle \langle e \rangle q$,
(iii) $\langle \theta^* \rangle q \equiv q$,
(iv) $\langle \theta \rangle q \equiv$ **false**.

Let $p'$ be the resultant formula. It is clear that any program occurring in $p'$ is either $A$ or $(A^n)^*$ for some $n \geq 1$. Note also that by construction, $p \equiv p'$ in $\mathcal{A}_m$.

We now give some brief definitions:

Define $L_A$ to be the subset of formulas of $\mathrm{PDL}_0$ in which the only programs occurring are $A$ or $(A^n)^*$ for $n \geq 1$.

For formulas $p \in L_A$, let $n_A(p)$ be the number of occurrences of $\langle A \rangle$.

## 4. Proof of main theorem

**Lemma 3.** *For any $p \in L_A$ and integers $m > 1$, $k \geq 0$ such that $2m+1$ is prime and $n_A(p) < m-k$,*

$$\mathcal{A}_m, w_k \models p \Leftrightarrow \mathcal{A}_m, w_{k+m} \models p. \tag{1}$$

**Proof.** Intuitively, the states $w_k$ and $w_{k+m}$ can only be distinguished if they force evaluation of propositional $P$ at $w_m$ and/or $w_{2m}$. However, the restrictions on $k$ and $m$ ensure that there are insufficient occurrences of $\langle A\rangle$ in $p$ to force such an evaluation. In addition, application of $\langle (A^n)^*\rangle$ obliterates any distinction between $w_k$ and $w_{k+m}$ by taking any state to any other state.

We proceed by induction on $|p|$. (Here $|p|$ denotes the length of $p$ regarded as a string over the alphabet $\Sigma_0 \cup \Phi_0 \cup \{\vee, \neg, \langle, \rangle, *, \cup, ;, \textbf{true}, \textbf{false}\}$.)

If $|p| = 1$ then $p$ is a propositional variable or **true** or **false** and for $0 \leq k < m$,

$$\mathcal{A}_m, w_k \models p \Leftrightarrow \mathcal{A}_m, w_{k+m} \models p.$$

Now assume (1) holds for $|p| < n$ and consider $|p| = n$. In each of the following cases let $m$, $k$ satisfy the conditions of (1) for $p$.

Let $p = \neg q$. Clearly, $|q| < |p|$ and $n_A(p) = n_A(q)$. By induction

$$\mathcal{A}_m, w_k \models q \Leftrightarrow \mathcal{A}_m, w_{k+m} \models q$$

and so

$$\mathcal{A}_m, w_k \models \neg q \Leftrightarrow \mathcal{A}_m, w_{k+m} \models \neg q.$$

Let $p = q \vee r$. $n_A(q) \leq n_A(p)$, $n_A(r) \leq n_A(p)$, $|q| < |p|$ and $|r| < |p|$, hence by induction,

$$\mathcal{A}_m, w_k \models q \Leftrightarrow \mathcal{A}_m, w_{k+m} \models q,$$

$$\mathcal{A}_m, w_k \models r \Leftrightarrow \mathcal{A}_m, w_{k+m} \models r.$$

It follows immediately that

$$\mathcal{A}_m, w_k \models q \vee r \Leftrightarrow \mathcal{A}_m, w_{k+m} \models q \vee r.$$

Let $p = \langle A\rangle q$. $n_A(q) = n_A(p) - 1 < m - (k+1)$ and $|q| < |p|$ hence by induction

$$\mathcal{A}_m, w_{k+1} \models q \Leftrightarrow \mathcal{A}_m, w_{k+1+m} \models q.$$

But then,

$$\mathcal{A}_m, w_k \models \langle A\rangle q \Leftrightarrow \mathcal{A}_m, w_{k+1} \models q$$
$$\Leftrightarrow \mathcal{A}_m, w_{k+1+m} \models q$$
$$\Leftrightarrow \mathcal{A}_m, w_{k+m} \models \langle A\rangle q.$$

Let $p = \langle (A^n)^* \rangle q$. If $(2m+1) | n$ then $(w_k, w_j) \in A^n$ if and only if $j = k$. Hence

$$\mathcal{A}_m, w_k \models \langle (A^n)^* \rangle q \Leftrightarrow \mathcal{A}_m, w_k \models q.$$

Since $|q| < |p|$ the result follows by induction. If $(2m+1) \nmid n$ then since $2m+1$ is prime,

$$0, n, 2n, 3n, \ldots, (2m)n$$

have all the $(2m+1)$ different residues modulo $(2m+1)$. Hence,

$$(w_k, w_j) \in (A^n)^* \quad \text{for all } w_j \in W^m$$

and

$$\mathcal{A}_m, w_k \models \langle (A^n)^* \rangle q \Leftrightarrow (\exists_j) \mathcal{A}_m, w_j \models q$$
$$\Leftrightarrow \mathcal{A}_m, w_{k+m} \models \langle (A^n)^* \rangle q.$$

We now can prove the

**Main Theorem.** *Let $z$ be the PDL formula $\langle (P?;A)^*, \neg P?;A;P?\rangle$***true***. There is no formula in* PDL$_0$ *equivalent to $z$.*

**Proof.** Suppose $p \in \mathrm{PDL}_0$ is equivalent to $z$. By Reduction Lemmas 1 and 2, there is a formula $p' \in L_A$ such that $p \equiv p'$ in $\mathcal{A}_m$ for all $m$. Let $k$, $m$ satisfy the conditions of Lemma 3, then

$$\mathcal{A}_m, w_k \models p' \Leftrightarrow \mathcal{A}_m, w_{k+m} \models p'.$$

But this contradicts the observation that

$$\mathcal{A}_m, w_0 \models z \quad \text{and} \quad \mathcal{A}_m, w_m \not\models z.$$

## 5. Conclusions and extensions

The proof given in Sections 1–4 actually shows that PDL$_1$ (PDL with at most one level of nesting of conditionals) is more expressive than PDL$_0$. This is a stronger result than that PDL is more expressive than PDL$_0$. Generally, for $n > 0$, define

$$\mathrm{PDL}_{n+1} = \{p \in \mathrm{PDL} \mid \text{if } q? \text{ is a subprogram in } p \text{ then } q \in \mathrm{PDL}_i \text{ for some } i \leq n\},$$

$$\mathrm{PDL} = \bigcup_n \mathrm{PDL}_n.$$

In addition, denote the statement "Language $L_1$ is less expressive than language $L_2$" by $L_1 \prec L_2$.

Having shown that $\mathrm{PDL}_0 \prec \mathrm{PDL}_1$, it is natural to ask whether for all $n \geq 0$, $\mathrm{PDL}_n \prec \mathrm{PDL}_{n+1}$. Using a distinguishing formula, cyclic model and proof techniques which generalize the formula $z$, model $\mathscr{A}_m$ and proof techniques in our original report [1], Peterson [5] and Berman [2] independently showed that for any $n \geq 0$, $\mathrm{PDL}_n \prec \mathrm{PDL}_{n+1}$. We briefly outline the argument given in [2].

(1) First define a formula $z_{n+1}$ in $\mathrm{PDL}_{n+1}$ and a class of cyclic models $\{\mathscr{A}_{m_n, m_{n-1}, \ldots, m_0} = \mathscr{A}_{\bar{m}}\}$. Assume towards a contradiction that $p$ is a formula in $\mathrm{PDL}_n$ equivalent to $z_{n+1}$.

(2) Reduce $p$ to a formula $p'$ in normal form. As before $p$ and $p'$ will be equivalent in each $\mathscr{A}_{\bar{m}}$ with $m_i > 1$ for $0 \leq i \leq n$.

(3) Show that for any $p'$ in normal form, and "large enough" model $\mathscr{A}_{\bar{m}}$ and particular states $w_{n,k_n}$ and $w_{n,k_n+m_n}$, $p'$ is satisfiable at $w_{n,k_n}$ in $\mathscr{A}_{\bar{m}}$ iff $p'$ is satisfiable at $w_{n,k_n+m_n}$ in $\mathscr{A}_{\bar{m}}$.

(4) This provides a contradiction since for similar $\bar{m}$ and $\bar{k}$, $z_{n+1}$ is satisfiable at $w_{n,k_n}$ in $\mathscr{A}_{\bar{m}}$ but not at $w_{n,k_n+m_n}$ in $\mathscr{A}_{\bar{m}}$.

The formula $z_{n+1}$ is based on our formula $z$ and defined recursively:

$$z_1 = \langle A_0; (P?;A_0)^*; \neg P?;A_0;P?\rangle\, \mathbf{true},$$

$$z_{n+1} = \langle A_n; (z_n?;A_n)^*; \neg z_n?;A_n;z_n\rangle\, \mathbf{true}.$$

The cyclic model $\mathscr{A}_{\bar{m}}$ consists of a sequence of copies of our model $\mathscr{A}_m$. Each state at level $i+1$ goes to either state $w_{i,1}$ or state $w_{i,m_i+1}$ at level $i$. For example, the model $\mathscr{A}_{4,4,4}$ can be embedded in Fig. 2. More formally, $\mathscr{A}_{\bar{m}} = (W, \Pi, \rho)$ where

$$W = \{w_{0,0}, \ldots, w_{0,2m_0-1}, w_{1,0}, \ldots, w_{1,2m_1-1}, \ldots, w_{n,0}, \ldots, w_{n,2m_n-1}\},$$

$$\begin{aligned}\rho(A_i) = {} & \{(w_{i,j}, w_{i,j+1}) \mid 0 \leq j < 2m_i - 1\} \\ & \cup \{(w_{i,2m_i}, w_{i,0})\} \\ & \cup \{(w_{i+1,j}, w_{i,1}) \mid j \neq m_{i+1}-2, 2m_{i+1}-2 \text{ or } 2m_{i+1}-1\} \\ & \cup \{(w_{i+1,j}, w_{i,m_i+1})\} \mid j = m_{i+1}-2, 2m_{i+1}-2 \text{ and } 2m_{i+1}-1\} \\ & \quad \text{for } 0 \leq i \leq n,\end{aligned}$$

$$\rho(b) = \emptyset \quad \text{for } b \neq A_0, \ldots, A_n,$$

$$\Pi(P) = W - \{w_{0,m_0-2}, w_{0,2m_0-2}, w_{0,2m_0-1}\},$$

$$\Pi(Q) = W \quad \text{for } Q \in \Sigma_0 - \{P\}.$$

For any state $w_{i,j}$, $i$ refers to the $i$th level or cycle and $j$ refers to the $j$th location within that cycle.

Essentially, $\mathrm{PDL}_n \prec \mathrm{PDL}_{n+1}$ is proved by showing that states at any level can only be distinguished by formulas which trace a path in the model to the bottom

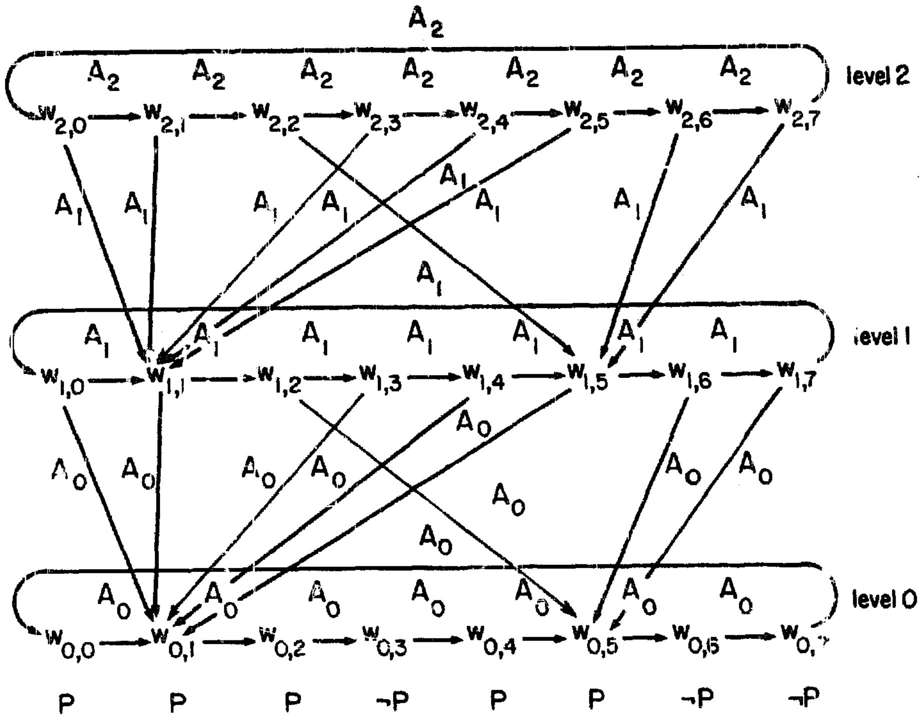


Fig. 2.

level (where states can be distinguished propositionally). To distinguish states at the top level, such formulas must have test depth at least $n+1$. Formulas with any smaller test depth cannot trace down deep enough in the model. For further details, we refer the reader to [2] or [5].


## Acknowledgment

The authors wish to thank Mike Fischer for helpful discussions and his support in this project, and Albert Meyer and the referees of Theoretical Computer Science for valuable suggestions.


## References


[1] F. Berman and M. Paterson, Test-free Propositional Dynamic Logic is Strictly weaker than PDL, Technical Report #77-10-02, University of Washington, Seattle, WA (1977).

[2] F. Berman, Expressiveness hierarchy for PDL with rich tests, Technical Report #78-11-01, University of Washington, Seattle, WA (1978).

[3] M. Fischer and R. Ladner, Propositional Dynamic Logic of programs, *J. Comput. System Sci.* **18** (1979).

[4] S. Ginsburg, *The Mathematical Theory of Context-Free Languages* (McGraw-Hill, New York, 1966) 86.

[5] G. Peterson, The power of tests in Propositional Dynamic Logic, Technical Report #47, University of Rochester, Rochester, NY.