A Hierarchical Analysis of Propositional Temporal Logic based on Intervals

Ben Moszkowski

Software Technology Research Laboratory

De Montfort University

Leicester

Great Britain

email: **`x@y`**, where **`x=benm`** and **`y=dmu.ac.uk`**

`http://www.cse.dmu.ac.uk/~benm`

Introduction

- We present a new hierarchical framework for analysing *Proposition Temporal Logic* (PTL).

- Our approach uses reasoning based on intervals of time.

- We obtain standard results such as a small model property, decision procedures and axiomatic completeness.

- Both finite time and infinite time are considered.

- Analyse PTL with both the operator until and past time by reduction to a version of PTL without either one.

- Show useful links between PTL and Propositional Interval Temporal Logic (PITL).

Relevance Beyond PTL

- Significant application of ITL and interval-based reasoning.
- Illustrates general approach to formally reasoning about various issues involving discrete linear time (e.g., sequential and parallel composition).
- The formal notational framework hierarchically reduces infinite-time reasoning to simpler finite-time reasoning.
- Approach could be used in model checking.
- The work includes some interesting representation theorems.
- Uses fixpoints of a certain interval-oriented temporal operator.
- Relevant to hardware description and verification: Property specification languages **PSL/Sugar** (IEEE standard 1850) and **’temporal e’** (part of IEEE candidate standard 1647) contain constructs involving intervals of time.

Some Background

- Several analyses of PTL already exist (e.g., Gabbay et al., 1980).
- Common features of previous approaches:
  - Explicit representation of individual states as sets of formulas.
  - Canonical linear model of such sets.
  - Intermediate graphs with nodes which are sets of formulas.

  Exceptions:

  Vardi and Wolper ('86): Decision procedure using ω-automata.

  Lange and Stirling (LICS '01): Game theory.
- Lichtenstein and Pnueli ('00) give a detailed analysis of PTL which is meant to largely subsume and supercede earlier ones:

  "The paper summarizes work of over 20 years and is intended to provide a definitive reference to the version of propositional temporal logic used for the specification and verification of reactive systems."

Benefits of Our Approach

- Natural hierarchical framework using intervals of time.
  The operator $until$ and past time are “add-ons”.
- Provides logic for articulating issues in analysis of PTL.
- Reduction of infinite-time reasoning to finite-time reasoning.
- Direct construction from finite-length state sequences (intervals).
- Avoids graphs involving many sets of formulas, paths, etc.
- Suggests easy-to-describe BDD-based PTL decision procedure.
- Exploits axiomatic completeness of PTL subset with only $\bigcirc$.
- Reveals useful links between intervals, PTL, Propositional Interval Temporal Logic (PITL) and fixpoints of interval-based operators.

A companion paper (JANCL ’04) gives completeness proof for PITL with finite time by a similar hierarchical reduction to PTL.

Structure of Presentation

- Introduction
- **Review of PTL and intervals**

Propositional Temporal Logic

Popular logic for specifying and verifying properties of time.

Has tool support widely used in academia and industry.

1996 ACM Turing award given to Prof. Amir Pnueli:

> “For his seminal work introducing temporal logic into computing science and for outstanding contributions to program and system verification.”

PTL Syntax

In what follows, p is any propositional variable and both X and Y themselves denote PTL formulas:

$$p \qquad \mathit{true} \qquad \neg X \qquad X \vee Y$$

$\bigcirc X$ (“next X”)

$\Diamond X$ (“eventually X”).

Variables such as X, X' and Y normally denote arbitrary PTL formulas.

No until operator or past time.

Derive other Boolean constructs: false, $X \wedge Y$, $X \supset Y$ and $X \equiv Y$.

Intervals of Time

Discrete, linear time is represented by **intervals** (i.e., sequences of states).

An **interval** σ consists of either

- a finite, nonzero number of states $\sigma_0, \sigma_1, \ldots$.
- or infinite (i.e., ω) states.

Each state σ_i maps each variable p, $q, \ldots$ to $true$ or $false$.

The value of p in the state σ_i is denoted $\sigma_i(p)$.

Semantics of PTL

The notation $\sigma \models X$ denotes that the interval σ satisfies the PTL formula X. Below is the semantics of the basic PTL constructs:

- $\sigma \models p$ iff $\sigma_0(p) = true$. (Use p's value in σ's initial state σ_0)
- $\sigma \models true$ trivially holds for any σ.
- $\sigma \models \neg X$ iff $\sigma \not\models X$.
- $\sigma \models X \vee Y$ iff $\sigma \models X$ or $\sigma \models Y$.
- $\sigma \models \bigcirc X$ iff σ has at least 2 states and $\sigma' \models X$, where σ' denotes $\sigma_1\sigma_2\ldots$.
- $\sigma \models \Diamond X$ iff for some suffix σ' of σ, $\sigma' \models X$.

Sample PTL Formulas and Intervals

$p \wedge \bigcirc(\neg p \wedge \bigcirc \neg p)$

	•	•	•
p:	t	f	f

$p \wedge \bigcirc\bigcirc\neg\bigcirc \mathit{true}$

	•	•	•
p:	t	f	t

$\neg p \wedge q$
$\quad \wedge \Diamond(p \wedge \neg q)$

	•	•	•	•
p:	f	t	t	f
q:	t	t	f	f

$\Diamond(\neg p \wedge \bigcirc p)$

	•	•	•	•	•	•
p:	f	t	t	f	t	f

$\neg \Diamond \neg p \quad (\Box\, p)$

	•	•	•	•	•	•
p:	t	t	t	t	t	t

Satisfiability and Validity

If $\sigma \models X$ for some σ, then X is **satisfiable**.

If $\sigma \models X$ for all σ, then X is **valid**.

Derived PTL operator $\Box$:

$$\Box X \quad \overset{\mathrm{def}}{\equiv} \quad \neg \Diamond \neg X \qquad \text{(Henceforth)}$$

Hierarchical Analysis without Past Time

Full PTL without past time (e.g., $\Box\Diamond p \wedge \Box\Diamond\neg p$)

$$\Downarrow$$

Invariant configurations in PTL (without past time)

(e.g., $\Box I \wedge w$, with

$$\begin{aligned} I: \quad & (r_1 \equiv \Diamond p) \wedge (r_2 \equiv \Diamond\neg r_1) \\ & \wedge (r_3 \equiv \Diamond\neg p) \wedge (r_4 \equiv \Diamond\neg r_3) \\ w: \quad & \neg r_2 \wedge \neg r_4 \end{aligned}$$)

$$\Downarrow$$

Transition configurations in PTL (without past time)

(e.g., $\Box T \wedge w \wedge \mathit{finite}$ (finite defined shortly), with

$$\begin{aligned} T: \quad & (r_1 \equiv (p \vee \bigcirc r_1)) \wedge (r_2 \equiv (\neg r_1 \vee \bigcirc r_2)) \\ & \wedge (r_3 \equiv (\neg p \vee \bigcirc r_3)) \wedge (r_4 \equiv (\neg r_3 \vee \bigcirc r_4)) \\ w: \quad & \neg r_2 \wedge \neg r_4 \end{aligned}$$)

$$\Downarrow$$

Low-level formulas in PITL

More Operators Definable in PTL

(Most concern finite time and are not well known)

more	$\overset{\mathrm{def}}{\equiv}$	$\bigcirc\, \mathit{true}$	More than one state
empty	$\overset{\mathrm{def}}{\equiv}$	$\neg \mathit{more}$	Only one state (*empty interval*)
skip	$\overset{\mathrm{def}}{\equiv}$	$\bigcirc\, \mathit{empty}$	Exactly two states (*unit interval*)
$\$\, X$	$\overset{\mathrm{def}}{\equiv}$	$X \wedge \mathit{skip}$	Unit interval with test (*unit test*)
finite	$\overset{\mathrm{def}}{\equiv}$	$\Diamond\, \mathit{empty}$	Finite interval
inf	$\overset{\mathrm{def}}{\equiv}$	$\neg \mathit{finite}$	Infinite interval
$\mathit{fin}\ X$	$\overset{\mathrm{def}}{\equiv}$	$\Box(\mathit{empty} \supset X)$	Weak test of final state
$\boxed{\mathrm{m}}\ X$	$\overset{\mathrm{def}}{\equiv}$	$\Box(\mathit{more} \supset X)$	“Mostly” (Henceforth before end.)

More Sample PTL Formulas and Intervals

Recall: $\mathit{more} \overset{\text{def}}{\equiv} \bigcirc \mathit{true}$ $\quad \mathit{empty} \overset{\text{def}}{\equiv} \neg \mathit{more}$ $\quad \mathit{skip} \overset{\text{def}}{\equiv} \bigcirc \mathit{empty}$

$\$\, X \overset{\text{def}}{\equiv} X \wedge \mathit{skip}$ $\quad \boxed{\text{m}}\, X \overset{\text{def}}{\equiv} \Box(\mathit{more} \supset X)$

$$\mathit{skip} \wedge \mathit{fin}\, \neg p$$

	•	•
p:	t	f

$$\bigcirc \$(p \supset \bigcirc \neg p) \\ \wedge \neg \$(p \wedge \bigcirc p)$$

	•	•	•
p:	t	t	f

$$\boxed{\text{m}}(p \supset \bigcirc \neg p) \\ \wedge \neg \Box(p \supset \bigcirc \neg p)$$

	•	•	•	•	•	•
p:	t	f	t	f	f	t

$$\boxed{\text{m}}(p \supset \Diamond \neg p) \\ \wedge \mathit{fin}\, p$$

	•	•	•	•	•	•
p:	t	t	t	t	f	t

Use $\boxed{\text{m}}$ instead of $\Box$ to reason about pairs of adjacent states without "running off end" of finite intervals. (See later Theorem 1.)

Some Conventions for Variables

- V denotes the finite set of propositional variables used.

- w and w' denote **state formulas**, i.e., ones without temporal operators.

- The set of PTL formulas in which the only primitive temporal operator is $\bigcirc$ is called **Next Logic** (NL).

  The subset of NL with no $\bigcirc$ nested in another $\bigcirc$ is denoted NL^1.

  Example: The NL formula $p \wedge \bigcirc q$ is in NL^1, but the NL formula $p \wedge \bigcirc(q \vee \bigcirc p)$ is not.

- T, T' and T'' denote formulas in NL^1.

Atoms

An **atom** is any finite conjunction in which each conjunct is some propositional variable or its negation and no two conjuncts share the same variable.

Example: $p \wedge \neg q$ is an atom but $p \wedge \neg p$ is not.

For any finite set of propositional variables V, let Atoms$_V$ be some set of $2^{|V|}$ logically distinct atoms containing exactly the variables in V.

Example: Four logically distinct atoms in Atoms$_{\{p,q\}}$:

$$p \wedge q \quad p \wedge \neg q \quad \neg p \wedge q \quad \neg p \wedge \neg q.$$

The Greek letters α and β denote individual atoms in Atoms$_V$.

- Introduction
- Review of PTL and intervals
- **Propositional Interval Temporal Logic (PITL)**

Features of Interval Temporal Logic (ITL)

- Modular reasoning about time (e.g., hardware, multimedia)
- Flexible notation for discrete linear order
- Supports sequential operators found in programs, etc.
- Compositionality with assumptions and commitments
- Supports reasoning about both automata and regular expressions
- Hybrid systems: Duration Calculus
- Temporal projection
- ITL influenced Verisity Ltd.'s language **temporal e** (part of candidate IEEE standard 1647). Verisity has now been acquired by Cadence Design Systems, Inc., a leading supplier of electronic design technologies and engineering services.

Syntax of PITL

All PTL constructs are permitted as well as two new ones.

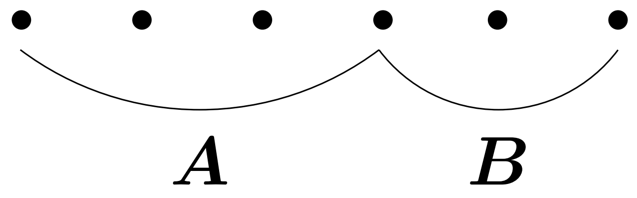
Here is the syntax of PITL's two extra primitive constructs, where $\boldsymbol{A}$ and $\boldsymbol{B}$ are themselves PITL formulas:

$$\boldsymbol{A};\boldsymbol{B}\ \ (\textit{chop}) \qquad\qquad \boldsymbol{A}^*\ \ (\textit{chop-star}).$$

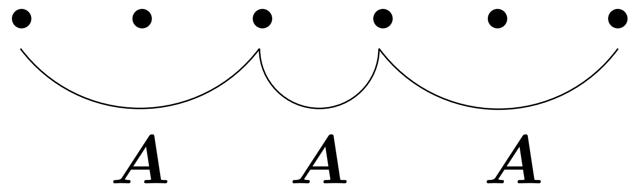
Semantics of PITL for Finite Time

The same kind of discrete-time intervals as in PTL.

$A; B$



A^*



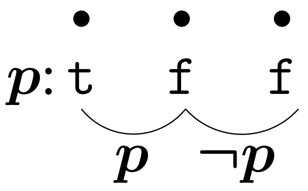
Each pair of adjacent subintervals share a state.

Sample PITL Formulas with Finite Time

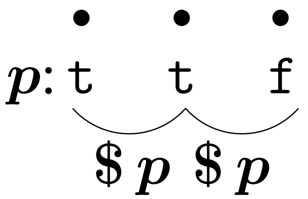
Recall: $\mathit{more} \overset{\mathrm{def}}{\equiv} \bigcirc\, \mathit{true}$ $\quad$ $\mathit{empty} \overset{\mathrm{def}}{\equiv} \neg \mathit{more}$ $\quad$ $\mathit{skip} \overset{\mathrm{def}}{\equiv} \bigcirc\, \mathit{empty}$

$\mathit{finite} \overset{\mathrm{def}}{\equiv} \Diamond\, \mathit{empty}$ $\quad$ $\$\, X \overset{\mathrm{def}}{\equiv} X \wedge \mathit{skip}$

$p; \neg p$

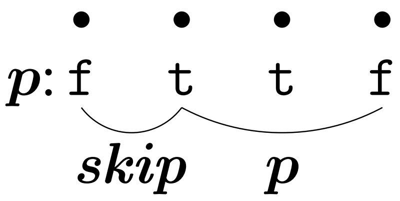


$(\$\, p)^*$



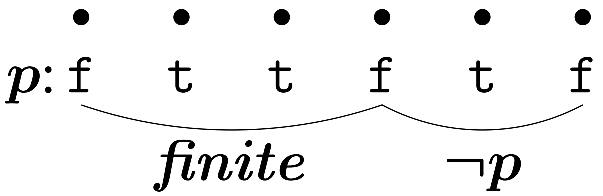
$\mathit{skip}; p$

$(\bigcirc p)$

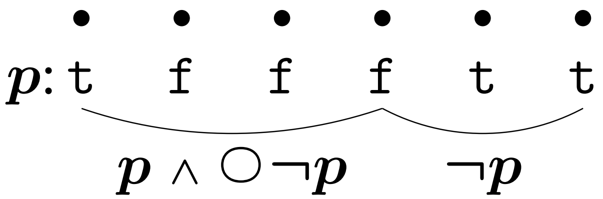


$\mathit{finite}; \neg p$

$(\Diamond \neg p)$



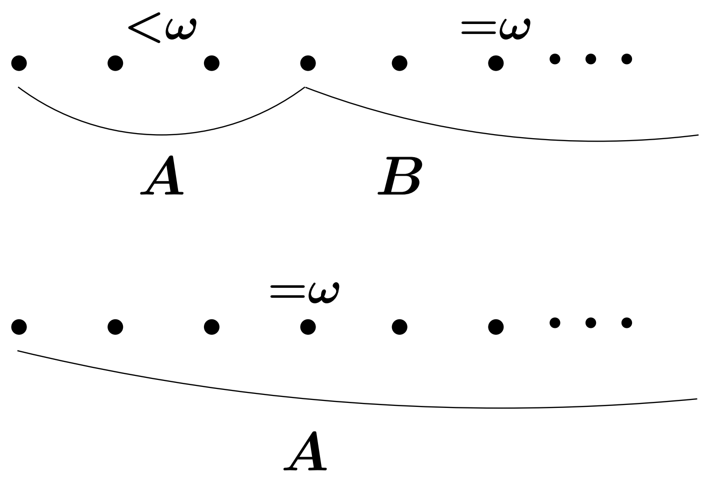
$(p \wedge \bigcirc \neg p); \neg p$



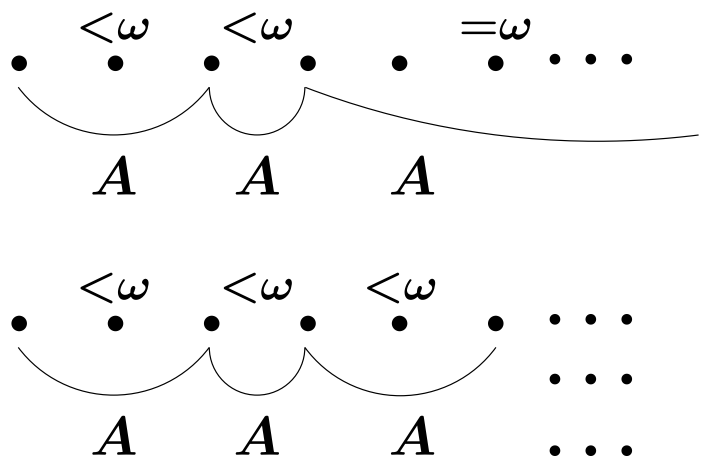
Semantics of PITL for Infinite Time

Extend *chop* and *chop-star* to include infinite time:

$A; B$



A^*



The Derived PITL Operator Chop-Omega

Define the next PITL operator called *chop-omega*:

$$A^{\omega} \quad \overset{\text{def}}{\equiv} \quad (A \wedge \mathit{finite})^{*} \wedge \mathit{inf}$$

Caution: Interval-oriented reasoning in PITL and PTL with finite time is different from conventional point-based reasoning in PTL with infinite-time.

- Introduction
- Review of PTL and intervals
- Propositional Interval Temporal Logic (PITL)
- **Transition configurations**

Recall Hierarchical Analysis without Past Time

Full PTL without past time

$\Downarrow$

Invariant configurations in PTL (without past time)

$\Downarrow$

Transition configurations in PTL **(without past time)**

$\Downarrow$

Low-level formulas in PITL

We obtain standard results such as a small model property, decision procedures and axiomatic completeness.

☞First analyse *Transition Configurations*.
They have simple syntax and yet capture essence of analysis.

Transition Configurations & Conditional Liveness Formulas

Four kinds of **Transition Configurations** (without past time):

Finite-time	$\Box T \wedge w \wedge \mathit{finite}$	
Infinite-time	$\Box T \wedge w \wedge \Box\Diamond^{+} L$	(Here $\Diamond^{+} X \overset{\text{def}}{\equiv} \bigcirc\Diamond X$)
Final	$\Box T \wedge w \wedge \mathit{empty}$	
Periodic	$\Box T \wedge \alpha \wedge L \wedge \Box\Diamond^{+}(\alpha \wedge L)$	(Recall $\alpha \in$ Atoms$_V$)

Here L is a **Conditional Liveness Formula** which is a conjunction of the form

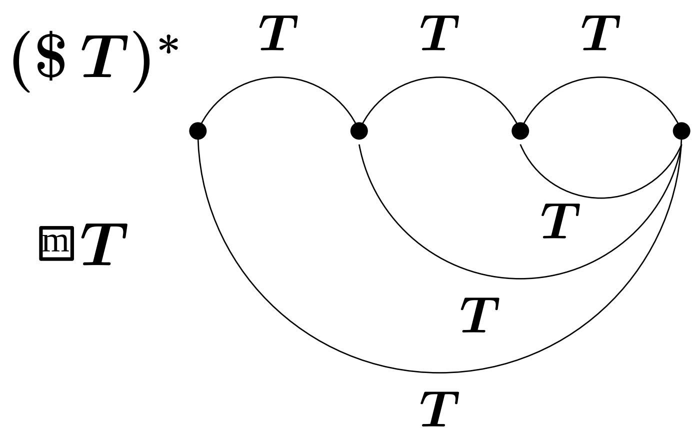
$$(w_1 \supset \Diamond w_1') \;\wedge\; (w_2 \supset \Diamond w_2') \;\wedge\; \cdots \;\wedge\; (w_{|L|} \supset \Diamond w_{|L|}').$$

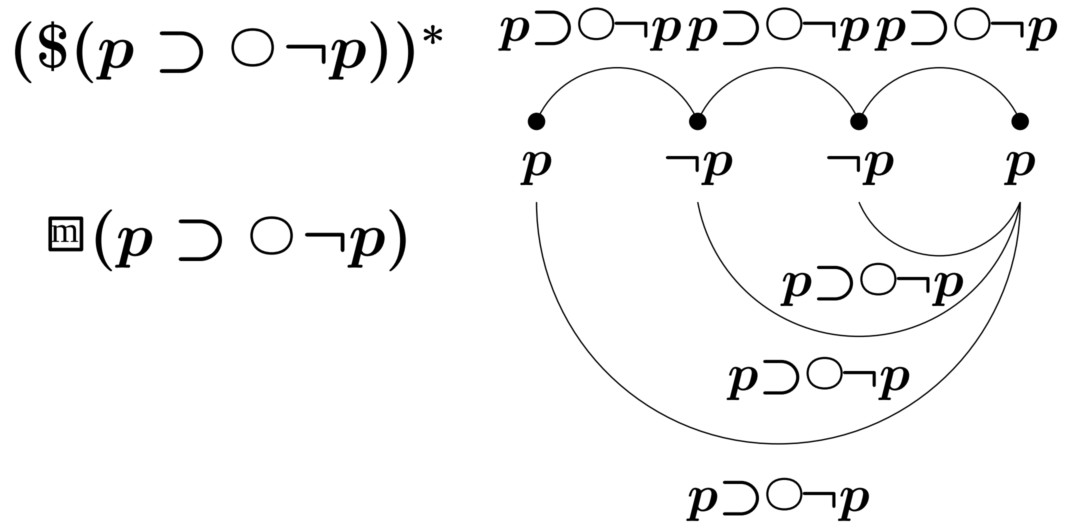
Expressing Transition Configurations with PITL

Theorem 1 *The* PITL *formula* $(\$\,T)^*$ *and the* PTL *formula* $\boxed{\text{m}}\,T$ *are semantically equivalent.*

Hence the next equivalence is valid: $(\$\,T)^* \equiv \boxed{\text{m}}\,T$.

Sample 4-state interval:





Reduction of Transition Configurations

Let $\vec{V} \leftarrow \vec{V}$ denote that initial and final values of variables in set V are equal.

Expressible in PTL: $\quad \mathit{finite} \supset \bigwedge_{v \in V}(v \equiv \mathit{fin}\, v)$.

Transition configuration	Equivalent PITL formula
$\Box T \wedge w \wedge \mathit{finite}$	$((\$T)^* \wedge w \wedge \mathit{finite}); (T \wedge \mathit{empty})$
$\Box T \wedge w \wedge \Box\Diamond^+ L$	$((\$T)^* \wedge w \wedge \mathit{finite});$ $\big((\$T)^* \wedge L \wedge (\vec{V} \leftarrow \vec{V})\big)^\omega$
$\Box T \wedge w \wedge \mathit{empty}$	$T \wedge w \wedge \mathit{empty}$
$\Box T \wedge \alpha \wedge L \wedge \Box\Diamond^+(\alpha \wedge L)$	$((\$T)^* \wedge \alpha \wedge L)^\omega$

Re-express $\Box T$ using $(\$T)^*$ (same as $\boxed{\mathrm{m}}\, T$ by Theorem 1).

The Operator $\Diamond\!\!\!\!\!\mathrm{f}\ $

For any PITL formula A, define new PITL construct $\Diamond\!\!\!\!\!\mathrm{f}\ A$:

$$\Diamond\!\!\!\!\!\mathrm{f}\ A \quad \overset{\mathrm{def}}{\equiv} \quad (A \wedge \mathit{finite}); \mathit{true}.$$

Intuition: $\Diamond\!\!\!\!\!\mathrm{f}\ A$ true on an interval σ iff A is true on some finite subinterval starting at the beginning of σ.

$\Diamond\!\!\!\!\!\mathrm{f}\ $-Fixpoints: A PITL formula A is a **fixpoint of $\Diamond\!\!\!\!\!\mathrm{f}\ $** iff the equivalence $A \equiv \Diamond\!\!\!\!\!\mathrm{f}\ A$ is valid.

Fixpoints of $\Diamond\!\!\!\!\!\mathrm{f}\ $ are easier to move in and out of subintervals than arbitrary formulas are.

Useful Theorem Concerning $\Diamond_f$-Fixpoints

The next theorem helps analyse periodic transition configurations:

Theorem 2 *For any $\Diamond_f$-fixpoint A, the next equivalence is valid:*

$$A \wedge \Box \Diamond^{+} A \quad \equiv \quad A^{\omega}.$$

We can use this to re-express periodic and infinite-time transition configurations in PITL.

Some Syntactic Categories of $\Diamond\!\!\!\!\!{\scriptstyle f}\;$-Fixpoints

Lemma 3 *Every state formula is a $\Diamond\!\!\!\!\!{\scriptstyle f}\;$-fixpoint.*
Furthermore, if the PITL *formulas A and B are $\Diamond\!\!\!\!\!{\scriptstyle f}\;$-fixpoints, then so are the following* PITL *formulas:*

$$A \wedge B \qquad A \vee B \qquad \bigcirc A \qquad \Diamond A.$$

Corollary 4 *If the* PITL *formula A is a $\Diamond\!\!\!\!\!{\scriptstyle f}\;$-fixpoint, so is $w \supset A$, where w is any state formula.*

Lemma 5 *Every conditional liveness formula L is a $\Diamond\!\!\!\!\!{\scriptstyle f}\;$-fixpoint.*

Proof: Recall that L has the following form:

$$w_1 \supset \Diamond w_1' \quad \wedge \quad \cdots.$$

Can therefore use Lemma 3 and Corollary 4.

$\Diamond\!\!\!\!\!\;{\scriptstyle f}\;$-Fixpoints and Periodic Transition Configurations

Recall Theorem 2: For any $\Diamond\!\!\!\!\!\;{\scriptstyle f}\;$-fixpoint A, have valid equivalence:

$$A \wedge \Box\Diamond^{+}A \quad\equiv\quad A^{\omega}.$$

Observe that $\alpha \wedge L$ is a $\Diamond\!\!\!\!\!\;{\scriptstyle f}\;$-fixpoint by Lemmas 3 and 5. Theorem 2 ensures the following valid equivalence:

$$\alpha \wedge L \wedge \Box\Diamond^{+}(\alpha \wedge L) \quad\equiv\quad (\alpha \wedge L)^{\omega}.$$

Then obtain following lemma:

Lemma 6 *The next equivalence concerning a periodic transition configuration is valid:*

$$\Box T \wedge \alpha \wedge L \wedge \Box\Diamond^{+}(\alpha \wedge L) \quad\equiv\quad ((\$T)^{*} \wedge \alpha \wedge L)^{\omega}. \tag{1}$$

Satisfiability for Periodic Transition Configurations

Theorem 7 *For any atom α in* Atoms$_V$*, the following are equivalent:*

(a) $\Box T \wedge \alpha \wedge L \wedge \Box \Diamond^+(\alpha \wedge L)$ *is satisfiable.*

(b) $\Box T \wedge \alpha \wedge L \wedge \Box \Diamond^+(\alpha \wedge L)$ *has a periodic model*

(by Lemma 6 *can use $((\$\, T)^* \wedge \alpha \wedge L)^\omega$).*

(c) *The next* PITL *formula is satisfiable in finite time:*

$(\$\, T)^* \wedge \alpha \wedge L \wedge \mathit{more} \wedge \mathit{finite} \wedge \mathit{fin}\, \alpha.$

☞ Shows reduction of infinite-time reasoning to finite-time reasoning.

☞ In (c), can replace $(\$\, T)^*$ with $\boxed{\text{m}}\, T$ to get PTL formula.

Satisfiability for Infinite-Time Transition Configurations

Theorem 8 *For any state formula w with variables in V, the following are equivalent:*

(a) $\Box T \wedge w \wedge \Box \Diamond^{+} L$ *is satisfiable.*

(b) $\Box T \wedge w \wedge \Box \Diamond^{+} L$ *has an ultimately periodic model (i.e., interval with periodic suffix).*

(c) The next PITL *formula is satisfiable in finite time:*
$(\$ T)^{*} \wedge w \wedge L \wedge \mathit{finite} \wedge \Diamond(\mathit{more} \wedge (\vec{V} \leftarrow \vec{V}))$.

☞ Shows reduction of infinite-time reasoning to finite-time reasoning.

☞ In (c), can replace $(\$ T)^{*}$ with $\boxed{m} T$ to get PTL formula.

- Introduction
- Review of PTL and intervals
- Propositional Interval Temporal Logic (PITL)
- Transition configurations
- **Small models for transition configurations**

Periodicity and Small Models

By Theorem 7, the periodic transition configuration $\Box T \wedge \alpha \wedge L \wedge \Box\Diamond^{+}(\alpha \wedge L)$ is satisfiable iff the next formula is satisfiable in finite time:

$$(\$\, T)^{*} \wedge \alpha \wedge L \wedge \mathit{more} \wedge \mathit{finite} \wedge \mathit{fin}\, \alpha.$$

Lemma 9 *If the formula $(\$\, T)^{*} \wedge \alpha \wedge L \wedge \mathit{more} \wedge \mathit{finite} \wedge \mathit{fin}\, \alpha$ is satisfiable, then it is satisfiable on a finite, nonempty interval having at most $(|L| + 1) \cdot |\mathrm{Atoms}_V|$ time units.*

Proof is by induction on $|L|$.

Use this to obtain small models for periodic and infinite-time transition configurations.

Small Models for Transition Configurations

Transition configuration	Upper bounds
$\Box\, T \ \wedge\ w \wedge \mathit{finite}$	Less than $\lvert\text{Atoms}_V\rvert$ units
$((\$\, T)^* \wedge w \wedge \mathit{finite}); (T \wedge \mathit{empty})$	
$\Box\, T \ \wedge\ w \wedge \Box\, \Diamond^+ L$	Initial part $< \lvert\text{Atoms}_V\rvert$, period $\leq (\lvert L\rvert + 1) \cdot \lvert\text{Atoms}_V\rvert$
$((\$\, T)^* \wedge w \wedge \mathit{finite}); \big((\$\, T)^* \wedge L \wedge (\vec{V} \leftarrow \vec{V})\big)^\omega$	
$\Box\, T \ \wedge\ w \wedge \mathit{empty}$	0 units (empty)
$T \wedge w \wedge \mathit{empty}$	
$\Box\, T \ \wedge\ \alpha \wedge L \wedge \Box\, \Diamond^+(\alpha \wedge L)$	Period $\leq (\lvert L\rvert + 1) \cdot \lvert\text{Atoms}_V\rvert$
$((\$\, T)^* \wedge \alpha \wedge L)^\omega$	

- Introduction
- Review of PTL and intervals
- Propositional Interval Temporal Logic (PITL)
- Transition configurations
- Small models for transition configurations
- **A BDD-based decision procedure**

A BDD-Based Decision Procedure: Case for Finite-Time Transition Configurations

Goal: Test $\Box T \wedge w \wedge \mathit{finite}$ for satisfiability:

Equivalent PITL formula: $((\$\,T)^* \wedge w \wedge \mathit{finite}); (T \wedge \mathit{empty})$.

This is satisfiable iff next three formulas are satisfiable for some atoms α and β in Atoms$_V$:

$$\alpha \wedge w \qquad (\$\,T)^* \wedge \alpha \wedge \mathit{finite} \wedge \mathit{fin}\,\beta \qquad T \wedge \beta \wedge \mathit{empty}.$$

Try to solve for such atoms α and β.

This can be done with **Symbolic State Space Traversal** techniques implemented using **Binary Decision Diagrams** (BDD).

A BDD-Based Decision Procedure: Case for Infinite-Time Transition Configurations

Goal: Test $\Box\, T \wedge w \wedge \Box\Diamond^{+} L$ for satisfiability:

Equivalent formula:

$$((\$\,T)^{*} \wedge w \wedge \mathit{finite});\big(((\$\,T)^{*} \wedge L \wedge (\vec{V} \leftarrow \vec{V}))\big)^{\omega}.$$

This is satisfiable iff next PTL formula satisfiable in finite-time:

$$\boxed{\text{m}}\, T \wedge w \wedge L \wedge \mathit{finite} \wedge \Diamond(\mathit{more} \wedge (\vec{V} \leftarrow \vec{V})). \qquad (2)$$

Can then do one of following:

- Apply finite-time decision procedure for full PTL.
- Reduce formula (2) directly to finite-time transition configuration.
- Utilise other BDD-based algorithms.

Sample Session of Prototype Implementation of the BDD-Based Decision Procedure

Goal: Test infinite-time satisfiability of $\Box\Diamond p \wedge \Box\Diamond\neg p$.

```
[6]> (dd-sat 'inf '(and (box (diamond (var p)))
                        (box (diamond (not (var p))))))
...
Satisfiable with infinite time.
...
Here is a model of an initial segment with 1 state:
***State 1:  P=1.
Here is a model of an (overlapping) periodic segment with
  3 states:
***State 1:  P=1.
***State 2:  P=0.
***State 3:  P=1.
...
[7]>
```

Corresponds to $p\,\neg p\,p\,\neg p\ldots$, i.e., the PITL formula$((\$\,p);(\$\,\neg p))^\omega$.

- Introduction
- Review of PTL and intervals
- Propositional Interval Temporal Logic (PITL)
- Transition configurations
- Small models for transition configurations
- A BDD-based decision procedure
- **Hierarchical analysis for full PTL without past time**

Hierarchical Analysis for Full PTL without Past Time

Full PTL without past time

$\Downarrow$

Invariant configurations (in PTL)

$\Downarrow$

Transition configurations (in PTL)

Example: Start with $\diamond p \wedge \square \diamond \neg p$.

Transform into a **invariant configuration** $\square I \wedge w$,

with $I: \; (r_1 \equiv \diamond p) \wedge (r_2 \equiv \diamond \neg r_3) \wedge (r_3 \equiv \diamond \neg p)$

$w: r_1 \wedge \neg r_2.$

Transform into a **finite-time transition configuration** $\square T \wedge w \wedge \mathit{finite}$,

with $T: \; (r_1 \equiv (p \vee \bigcirc r_1)) \wedge (r_2 \equiv (\neg r_3 \vee \bigcirc r_2))$

$\wedge \; (r_3 \equiv (\neg p \vee \bigcirc r_3))$

$w: r_1 \wedge \neg r_2.$

Hierarchical Analysis with Past Time

Full PTL with past time

$\Downarrow$

Invariant configurations with past time

$\Downarrow$

Transition configurations with past time

$\Downarrow$

Transition configurations without past time

Other Issues not Covered Here (Many in Paper)

- Axiomatic completeness
- Details of invariants and invariant configurations
- Reduce of arbitrary PTL formulas to invariants
- Treatment of the operator $until$ and past time
- Generalised conditional liveness formulas and invariants
- Fusion Logic with reduction to PTL (only finite time, not in paper)
- Some experience with using decision procedure (not in paper)
- Analysis of Propositional Dynamic Logic (PDL) without Fischer-Ladner closures (not in paper)
- Version of Hoare Logic with ITL pre- and post-conditions (not in paper)

Conclusions

- We have presented a new interval-based hierarchical framework for analysing PTL.
- It uses PITL to articulate various steps.
- It reduces infinite-time reasoning to finite-time reasoning.
- It complements existing methods.
- It complements our parallel work on a completeness proof for PITL using a hierarchical reduction to PTL via Fusion Logic.
- It suggests that the connection between PTL and PITL is more fundamental than generally considered.

Links to Paper and Information about ITL

- Paper at Computing Research Repository (CoRR):

  `http://arXiv.org/abs/cs.LO/0601008`

  Or go to following URL (e.g., Google search for CoRR):

  `http://arXiv.org/corr`

  Then search for Moszkowski or Interval Temporal Logic.

- Information on ITL, including downloadable book:

  `http://www.cse.dmu.ac.uk/STRL/ITL/`

  Or do web search (e.g., with Google) for

  ITL homepage or Interval Temporal Logic

(Extra Slides)

(Extra slides follow.)

Decomposition

Suppose $\alpha \in$ Atoms$_V$ and PITL formulas A and B have all variables in V.

Lemma 10 *The following are equivalent:*

- *The formula* $(A \wedge \mathit{finite}); (\alpha \wedge B)$ *is satisfiable.*
- *The two formulas* $A \wedge \mathit{finite} \wedge \mathit{fin}\, \alpha$ *and* $\alpha \wedge B$ *are satisfiable.*

Lemma 11 *The following are equivalent:*

- *The formula* $(\alpha \wedge A)^{\omega}$ *is satisfiable.*
- *The formula* $\alpha \wedge A \wedge \mathit{finite} \wedge \mathit{more} \wedge \mathit{fin}\, \alpha$ *is satisfiable.*