QF_BV Property Directed Reachability with Mixed Type Atomic Reasoning Units

Tobias Welp and Andreas Kuehlmann
1. Introduction

2. QF_BV Property Directed Reachability

3. Mixed Type Atomic Reasoning Units

4. Experimental Results

5. Summary
Motivation for Property Directed Reachability

- In 2011, Bradley proposed *Property Directed Reachability* (a.k.a IC\(^3\)) for model checking [Brad11].

- Experiments indicate that PDR outperforms model checking based on *Interpolation* [McMi03] on representative benchmark sets.

![Graph showing comparison between PDR and Interpolation](EénM11)
Other Favorable Properties of PDR

😊 No unrolling of transition relation.

😊 Parallizable.

😊 Allows for initialization with known invariants.

😊 Good for finding counterexamples and proving that none exists.
Research Pertaining PDR

- Ternary Simulation [EénM11]
- PDR with Non-State Variables [Back13]
- push-down systems [Hode12]
- WSTS [Kloo13]
- QF.LA [Hode12]
- QF_BV [Welp13]
- Timed systems [Kind12]
- Where Monolithic/Incremental Meet [Some11]
- Understanding IC3 [Brad12]
- Optimizations
- Generalization
- Analysis

Property Directed Reachability
Model Checking

• Given are
  – A set of initial states: $I(x)$
  – A set of bad states: $B(x)$
  – A transition relation: $T(x, x')$

• Question: Is a bad state reachable from an initial state using valid transitions?
Model Checking with PDR

Counterexample Sequence

- **init**
  - $x_1 = 1$
  - $x_2 = 0$

- **state 1**
  - $x_1 = 0$
  - $x_2 = 1$

- **bad**
  - $x_1 = 1$
  - $x_2 = 1$

PDR Trace

- **init**
  - 0

- 1

- 2

- 3
Proving a Safety Property with PDR

Legend:
- Initial set $I$
- Bad set $B$
- Proof oblig.
- Cover

- Can \textbf{bad} be reached within zero steps?
No, only the initial set is reachable within zero steps.

Everything else is covered, i.e. not reachable.
Can \textbf{bad} be reached within one step?

Conservatively, we initially assume that everything is \textit{reachable}.
Proving a Safety Property with PDR

• Find a point in bad that is not yet covered.
Proving a Safety Property with PDR

- Expand proof obligation using simulation.

Legend:
- Initial set $I$
- Bad set $B$
- Proof oblig.
- Cover
Proving a Safety Property with PDR

- The cube cannot be reached from the reachable area in frame 0.
Hence, we can consider the proof obligation covered.
Proving a Safety Property with PDR

- Expand the **covered** cube as much as possible.
Proving a Safety Property with PDR

- Repeat with finding a new point in bad that is not covered.
Proving a Safety Property with PDR

- Again, the point cannot be reached from the reachable area in the previous frame.
- Expand the covered cube.
- Now, bad is completely covered.
Can \text{bad} be reached within two steps?
Identified an inductive invariant disjoint from \textit{bad}.

This proves the property.
1. Introduction

2. QF\_BV Property Directed Reachability

3. Mixed Type Atomic Reasoning Units

4. Experimental Results

5. Summary
## Property Directed Reachability for QF$_{BV}$

<table>
<thead>
<tr>
<th>Original Formulation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Atomic Reasoning</th>
<th>Boolean Cubes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Expansion of Proof Obligations</th>
<th>Ternary Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Strengthen</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Weaknesses</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]
\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]
\[ B := (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]
Example Hybrid Invariant

\[ I \ := \ (2 \times y \equiv x) \land (x + y \leq 3) \]

\[ T \ := \ (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]

\[ B \ := \ (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]

Legend:
- Initial set \( I \)
- Bad set \( B \)
- Proof oblig.
- Cover
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]
\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]
\[ B := (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]
Example Hybrid Invariant

\[ I \ := \ (2 \times y \equiv x) \land (x + y \leq 3) \]
\[ T \ := \ (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]
\[ B \ := \ (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]
\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]
\[ B := (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]

\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]

\[ B := (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]

Legend:
- Initial set \( I \)
- Bad set \( B \)
- Proof oblig.
- Cover
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]

\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]

\[ B := (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]

Legend:
- **Initial set** \( I \)
- **Bad set** \( B \)
- **Proof oblig.**
- **Cover**
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]
\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]
\[ B := (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]
# Property Directed Reachability for QF\_BV

<table>
<thead>
<tr>
<th></th>
<th>Original Formulation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Atomic Reasoning</strong></td>
<td></td>
</tr>
<tr>
<td>Unit</td>
<td>Boolean Cubes</td>
</tr>
<tr>
<td></td>
<td>Ternary Simulation</td>
</tr>
<tr>
<td><strong>Expansion of</strong></td>
<td></td>
</tr>
<tr>
<td>Proof Obligations</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Strengths</strong></td>
<td>logic</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Weaknesses</strong></td>
<td>arithmetic</td>
</tr>
</tbody>
</table>
## Property Directed Reachability for QF\_BV

<table>
<thead>
<tr>
<th>Atomic Reasoning Unit</th>
<th>Boolean Cubes</th>
<th>Polytopes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expansion of Proof Obligations</td>
<td>Ternary Simulation</td>
<td>Interval Simulation</td>
</tr>
<tr>
<td>Strengths</td>
<td>logic</td>
<td></td>
</tr>
<tr>
<td>Weaknesses</td>
<td>arithmetic</td>
<td></td>
</tr>
</tbody>
</table>
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]
\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]
\[ B := (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]
\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]
\[ B := (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]

Legend:
- Initial set \( I \)
- Bad set \( B \)
- Proof oblig.
- Cover
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]

\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]

\[ B := (x + y \geq 4) \lor (x \text{ mod } 2 \equiv 1) \]
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]

\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]

\[ B := (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]
\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]
\[ B := (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]
\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]
\[ B := (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]
\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]
\[ B := (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]
\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]
\[ B := (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]
Example Hybrid Invariant

\[
I := (2 \times y \equiv x) \land (x + y \leq 3)
\]

\[
T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x)
\]

\[
B := (x + y \geq 4) \lor (x \mod 2 \equiv 1)
\]
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]
\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]
\[ B := (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]
## Property Directed Reachability for QF\_BV

<table>
<thead>
<tr>
<th></th>
<th>Original Formulation</th>
<th>Polytopes [Welp13]</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Atomic Reasoning</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unit</td>
<td></td>
<td>Boolean Cubes</td>
</tr>
<tr>
<td>Expansion of Proof Obligations</td>
<td></td>
<td>Polytopes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ternary Simulation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Interval Simulation</td>
</tr>
<tr>
<td><strong>Strengths</strong></td>
<td></td>
<td>logic</td>
</tr>
<tr>
<td></td>
<td></td>
<td>arithmetic</td>
</tr>
<tr>
<td><strong>Weaknesses</strong></td>
<td></td>
<td>arithmetic</td>
</tr>
<tr>
<td></td>
<td></td>
<td>logic</td>
</tr>
</tbody>
</table>
Outline

1. Introduction

2. QF\_BV Property Directed Reachability

3. Mixed Type Atomic Reasoning Units

4. Experimental Results

5. Summary
### Property Directed Reachability for QF\_BV

<table>
<thead>
<tr>
<th></th>
<th>Original Formulation</th>
<th>Polytopes [Welp13]</th>
<th>Hybrid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atomic Reasoning</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unit</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Expansion of Proof Obligations</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Strengths</td>
<td></td>
<td>logic</td>
<td>arithmetic</td>
</tr>
<tr>
<td>Weaknesses</td>
<td></td>
<td>arithmetic</td>
<td>logic</td>
</tr>
</tbody>
</table>

- **Unit Cube**
- **Boolean Cubes**
- **Polytopes**
- **Ternary Interval Simulation**
- **Hybrid Simulation**

January 23, 2014 ©Tobias Welp
Example Hybrid Invariant

\[
I := (2 \times y \equiv x) \land (x + y \leq 3)
\]

\[
T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x)
\]

\[
B := (x + y \geq 4) \lor (x \mod 2 \equiv 1)
\]

Legend:
- Initial set \(I\)
- Bad set \(B\)
- Proof oblig.
- Cover
Example Hybrid Invariant

\[ I := (2 \times y \equiv x) \land (x + y \leq 3) \]
\[ T := (y' \equiv y + 1) \land (x' \equiv x - 2) \land (y' > y) \land (x' < x) \]
\[ B := (x + y \geq 4) \lor (x \mod 2 \equiv 1) \]

Legend:
- Initial set \( I \)
- Bad set \( B \)
- Proof oblig.
- Cover
 Probabilistic Specialization

specialize to Boolean cube with probability $c$

specialize to polytope with probability $1 - c$
specialize to Boolean cube with probability $c$

specialize to polytope with probability $1 - c$
Probabilistic Specialization

specialize to Boolean cube with probability $c$

specialize to polytope with probability $1 - c$
The favorable event can be expected to happen in a constant number of steps.

\[ E\{\text{trials until Boolean cube specialization}\} = c \sum_{i=1}^{\infty} i(1-c)^{i-1} = \frac{1}{c} \]

Analogously, one calculates

\[ E\{\text{trials until polytope specialization}\} = (1-c) \sum_{i=1}^{\infty} ic^{i-1} = \frac{1}{1-c} \]
Simulation-based expansion of proof obligations is e.g. used to expand a bad ARU that is not yet covered:
Simulation-based expansion of proof obligations is e.g. used to expand a bad ARU that is not yet covered:
Simulation-based Expansion

The check whether or not an expansion is valid can be reduced to simulation on sets of points [EénM11].

Assume \( \text{bad} \) is defined as \( e_1 < 2 \) and we already covered \( e_1 < -1 \) with

\[
e_1 := (x_1 - x_2 + 2) \land (y_1 \lor y_2)
\]

Then we may expand an ARU \( a \) to a larger ARU \( \hat{a} \) if \( e := (e_1 < 2) \land \neg (e_1 < -1) \) evaluates to \text{true} for all values in \( \hat{a} \).
Let \( \hat{a} := (1 \leq x_1 \leq 5) \land (0 \leq x_2 \leq 3) \land (y_1 \in -00-) \land (y_2 \in 100-) \)
Ternary Simulation

Let $\hat{a} := (1 \leq x_1 \leq 5) \land (0 \leq x_2 \leq 3) \land (y_1 \in -00-) \land (y_2 \in 100-) \land \Phi$ Ternary Simulation
Let \( \hat{a} \) := \((1 \leq x_1 \leq 5) \land (0 \leq x_2 \leq 3) \land (y_1 \in -00-) \land (y_2 \in 100-)\)
Let \( \hat{a} := (1 \leq x_1 \leq 5) \land (0 \leq x_2 \leq 3) \land (y_1 \in {-00-}) \land (y_2 \in {100-}) \)
The check whether or not an expansion is valid can be reduced to simulation on sets of points [EénM11].

Assume \( \text{bad} \) is defined as \( e_1 < 2 \) and we already covered \( e_1 < -1 \) with

\[
e_1 := (x_1 - x_2 + 2) \land (y_1 \lor y_2)
\]

Then we may expand an ARU \( \hat{a} \) to a larger ARU \( \hat{a} \) if \( e := (e_1 < 2) \land \neg (e_1 < -1) \) evaluates to \text{true} for all values in \( \hat{a} \).
Let $\hat{a} := (1 \leq x_1 \leq 5) \land (0 \leq x_2 \leq 3) \land (y_1 \in -00-) \land (y_2 \in 100-)$. 

\[ 
\begin{array}{c}
\Phi \text{ Ternary Simulation} \\
\Phi \text{ Interval Simulation} \\
\end{array}
\]
Let $\hat{a} := (1 \leq x_1 \leq 5) \land (0 \leq x_2 \leq 3) \land (y_1 \in -00-) \land (y_2 \in 100-)$
### Property Directed Reachability for QF\_BV

<table>
<thead>
<tr>
<th></th>
<th>Original Formulation</th>
<th>Polytopes [Welp13]</th>
<th>Hybrid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atomic Reasoning</td>
<td>Boolean Cubes</td>
<td>Polytopes</td>
<td>Boolean Cubes and Polytopes</td>
</tr>
<tr>
<td>Unit</td>
<td>Ternary Simulation</td>
<td>Interval Simulation</td>
<td>Hybrid Simulation</td>
</tr>
<tr>
<td>Expansion of Proof Obligations</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Strengths</td>
<td>logic</td>
<td>arithmetic</td>
<td>arithmetic logic</td>
</tr>
<tr>
<td>Weaknesses</td>
<td>arithmetic</td>
<td>logic</td>
<td>-</td>
</tr>
</tbody>
</table>
1. Introduction

2. QF_BV Property Directed Reachability

3. Mixed Type Atomic Reasoning Units

4. Experimental Results

5. Summary
Experimental Setup

```c
int foo
{
    while (x)
    {
        x--;  
    }
    assert (!x);
}
```
Benchmark Sets

Bitvector set of SV-Comp [Beye12]

```
int
foo(int n)
{
    int x = 1;
    while(1)
    {
        x += 2*n;
        assert(x);
    }
}
```

InvGen-Benchmarks [Gupt09]

```
int
foo(int n)
{
    int x = 0;
    assume(n>=0);
    while(x < n)
    {
        x--;  
    }
    assert(x <= n);
}
```

Mostly Logic Invariants

Mostly Arithmetic Invariants
Overall Performance

SV-COMP
[Beye12]

InvGen
[Gupt09]
Overall Performance

Number Solved Instances

\( c \)

[0s,0.5s] [0.5s,5s] [5s,50s] [50s,500s]
Impact of Simulation Type

SV-COMP
[Beye12]

InvGen
[Gupt09]
Impact of Simulation Type

All
Comparison vs ABC PDR

<table>
<thead>
<tr>
<th>Solving Time QF_BV PDR (s)</th>
<th>Solving Time ABC (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>10</td>
<td>100</td>
</tr>
</tbody>
</table>

timeout
Outline

1. Introduction

2. Property Directed Reachability

3. Generalization of PDR to QF_BV

4. Experimental Results

5. Summary
Summary

- PDR is an efficient algorithm for solving model checking problems.

- PDR with Boolean cubes performances poorly with arithmetic invariants.

- PDR with polytopes performances poorly with bit-level invariants.

- The hybrid formulation outperforms the pure versions.
Thank you!

for your

attention

twelp@berkeley.edu
References


References


