Adaptive Interpolation-Based Model Checking

Chien-Yu (Leo) Lai, Cheng-Yin Wu, Chun-Yang (Ric) Huang

2014.1.23
Outline

• Introduction
• Adaptive IMC Framework
• Flexible Interpolation
• Experimental Results
• Conclusion
INTRODUCTION
Interpolation-Based Model Checking (IMC)\(^1\)

\[ I_0 \land T \land \neg P \land k=0 \]

**SAT**: Increase \( k \)

**UNSAT**: \( I_0 \land T^k \land \neg P \) ?

**R = over-approximation within one more step**

\[ I_0 \land T^k \land R ? \]

**SAT**: Fixed point? No

**UNSAT**: Yes

\[^1\text{K. L. McMillan, Interpolation and SAT-based model checking (CAV 2003)}\]
Interpolation-Based Model Checking (IMC)

BMC Phase: $I_0^\wedge T^k \wedge \neg P$ ?

ITP Phase: Iteratively compute over-approximation step by step

Spurious counter-example

Real counter-example

UNSAFE

inconsistent

SAFE

Fixed point?

Refinement: increase $k$
Interpolation-Based Model Checking (IMC)

\[ I_0, T, !P, k=0 \]

Fast?

- Fast?
- Fast?
- Fast?

Increase k

SAT

UNSAT

UNSAFE

Adequate?

UNSAFE

SAFE

R = over-approximation

Fixed point?

Yes

No
Too fine-grained

\[ I_0, T, \neg P, k=0 \]

\[ I_0^{\wedge T^k \wedge \neg P} ? \]

**SAT**

**UNSAFE**

**UNSAT**

Increase \( k \)

\[ I_0^{\wedge T^k \wedge R} ? \]

**SAT**

\[ R = \text{over-approximation within one more step} \]

**SAFE**

**UNSAT**

Requires several iterations to jump out

Fixed point?

Yes

No

\[ I_0^{\wedge T^k \wedge \neg P} ? \]

**SAT**

**UNSAFE**

**UNSAT**
Too Coarse

I_0, T, !P, k=0

I_0^{T^k!P}?

Increase k

SAT

UNSAFEx

R = over-approximation within one more step

Hardly reach fixed point before spurious counter-examples

Need frequent refinement

I_0^{T^k!P}?

SAT

UNSAT

I_0^{T^k!P}?

No

Fixed point?

Yes

SAFE
Two examples

Need for finer-grained abstraction

Need for coarser abstraction

Spurious counter-example

Abstract reachability

Bad states
Previous Works – Single, Blind Granularities

• McMillan’s IMC\textsuperscript{1}
  – Depends only on the refutation proof

• NewITP\textsuperscript{2}
  – Depends only on the strength of SAT/UNSAT generalizations

\textsuperscript{1}K. L. McMillan, Interpolation and SAT-based model checking (CAV 2003)
\textsuperscript{2}C.Y. Wu, A counterexample-guided interpolant generation algorithm for SAT-based model checking (DAC’13)
Two examples (review)

With single granularity, IMC hardly solves both of them

Need for finer-grained abstraction

Need for coarser abstraction

Spurious counter-example

Abstract reachability

Bad states
ADAPTIVE IMC FRAMEWORK
Adaptive IMC Framework

\[ I_0^{\land T^k \land P} ? \]

- **SAT**: Increase \( k \)
- **UNSAT**: Fixed point? (No → Coarser, Yes → SAFE)

**Adaptivity**
- Tends to contain counter-examples → Finer
- Hard to Converge → Coarser

Flexible interpolation
FLEXIBLE INTERPOLATION BY REACHABILITY PARTITIONING
Reachability v.s. Granularity

• When $I_0^T T^{k-1} T^R$ is UNSAT, not all clauses get involved with UNSAT proof

- Concrete transitions
- Transitions by freed constrains
Reachability v.s. Granularity

- If the reachability is smaller, more clauses are absent in UNSAT proof
Make Abstraction Coarse

• By just partitioning \( R \) into 2 slices
Make Abstraction Coarse

- Constrains restricting the transitions from $R_1$ is missing
Make Abstraction Coarse

• Likewise

\[ l_0^{\wedge}T^{k-1} \]

---

Transitions by freed constrains
Make Abstraction Coarse

• The disjunction of the reachability becomes coarse than computing R’s directly

\[ I_0 \wedge T^{k-1} \]
Flexible Interpolation by Reachability Partitioning

\[ R_n \]

\[ \text{Reachability partitioner} \]

\[ n = \#\text{slices} \]

\[ r_1, r_n \]

\[ \text{ITP}_n \]

\[ \text{ITP}_1 \]

\[ \text{Disjoint} \]

\[ \text{Final ITP} \]
ATR&R INTERPOLATION
2-Step Interpolation

1. Transition Relation Abstraction

2. Reachability Construction
ATR to ATR Circuit

- Extract UNSAT core on the last time-frame

Extract this part
ATR Circuit

• Record the presence of clauses in proof
Ternary Simulation

• Finds don’t-care state variables

Find inputs For x0xx11x
ATR Circuit Simulation

• Similar to ternary simulation
• Consider constraints absent in abstract transition relation

\[(ab \rightarrow c)\]  
\[(c \rightarrow a)\]  
\[(c \rightarrow b)\]

\[c \text{ doesn’t imply } b \text{ anymore}\]
Interpolant Construction

• Iteratively Solve the previous states

After ATR circuit simulation
Adaptive IMC Framework (review)

I_0, T, !P, k=0

I_0^T_k^!P ?

Increase k

IO^T_k^!P ?

SAT

Increase k

IO^T_k^R ?

SAT

Hard to Converge ➔ Increase #slices

UNSAT

Fixed point?

Yes ➔ SAFE

No ➔ Unsafe

Tends to contain counter-examples ➔ Decrease #slices

ATR&R Interpolation

Adaptivity

FIRP

Adaptivity
What We Refine

• BMC step

• Interpolation Algorithm
EXPERIMENTAL RESULTS
Experiment Setup

• Intel(R) Xeon(R) CPU E5405, 2.00GHz
• 7GB memory, 15 minutes time-out
• hwmcc11nointel.7z
  – Downloaded from HWMCC website
• Initial number of slice: 1
  – Same as the McMillan’s IMC
Comparison in total cases

![Graph showing comparison in total cases for AIMC, NewItp, and McMillan.](image)

- Number of solved instances vs. time-out (sec)
- AIMC: 255
- NewItp: 244
- McMillan: 217
## Statistics in Detail

<table>
<thead>
<tr>
<th>405 cases in total</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIMC</td>
</tr>
<tr>
<td><strong>All Solved</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>Solved only</strong></td>
</tr>
<tr>
<td><strong>Unsolved only</strong></td>
</tr>
<tr>
<td><strong>All Unsolved</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>100 cases unsolved by PDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIMC</td>
</tr>
<tr>
<td><strong>Solved</strong></td>
</tr>
</tbody>
</table>
CONCLUSION
Contribution

• Adaptive interpolation framework

• Abstraction degree manipulation

• Enhancement of IMC
  – Solve the most instances in total
  – Solve the most instances hard for PDR
Novelty

- Flexible interpolation by reachability partitioning
- 2-phase interpolation
- 1-way SAT/UNSAT generalization by only one-time simulation
Thanks for Your Attention!