Interpolant Strength in Model Checking

Based on CAV'12 work

Simone Fulvio Rollini, O. Sery, N. Sharygina

Formal Verification Lab, University of Lugano

November 4th, 2012

Outline

1. Background
   - Interpolation for Model Checking
   - Labeled Interpolation Systems

Outline

1. Background
   - Interpolation for Model Checking
   - Labeled Interpolation Systems
2. Contribution
   - Interpolant Strength in Model Checking
   - Simultaneous Abstraction: Requirements and Constraints
   - Path Interpolation: Requirements and Constraints

Outline

1. Background
   - Interpolation for Model Checking
   - Labeled Interpolation Systems
2. Contribution
   - Interpolant Strength in Model Checking
   - Simultaneous Abstraction: Requirements and Constraints
   - Path Interpolation: Requirements and Constraints
3. Conclusions

Outline

1. Background
   - Interpolation for Model Checking
   - Labeled Interpolation Systems

2. Contribution
   - Interpolant Strength in Model Checking
   - Simultaneous Abstraction: Requirements and Constraints
   - Path Interpolation: Requirements and Constraints

3. Conclusions

Outline

1. Background
   - Interpolation for Model Checking
   - Labeled Interpolation Systems
2. Contribution
   - Interpolant Strength in Model Checking
   - Simultaneous Abstraction: Requirements and Constraints
   - Path Interpolation: Requirements and Constraints
3. Conclusions

- Model Checking

- Model Checking
    - System model vs behavioral property specification

Symbolic Model Checking

Hardware and Software Verification

- Model Checking
  - System model vs behavioral property specification
- Symbolic SAT-based approach

Symbolic Model Checking

Hardware and Software Verification

- Model Checking
  - System model vs behavioral property specification
- Symbolic SAT-based approach
  - System and properties as logic formulae

Symbolic Model Checking

Hardware and Software Verification

- Model Checking
  - System model vs behavioral property specification
- Symbolic SAT-based approach
  - System and properties as logic formulae
  - Problem encoding into logic (SAT)

- Model Checking
  - System model vs behavioral property specification
- Symbolic SAT-based approach
  - System and properties as logic formulae
  - Problem encoding into logic (SAT)
  - Problem solving by means of reasoning engines (SAT solvers)

- Bounded model checking: approximate reachability set computation [McM03]

- Bounded model checking: approximate reachability set computation [McM03]

- Predicate abstraction refinement based on spurious behaviors [HJRM04]

- Bounded model checking: approximate reachability set computation [McM03]
- Predicate abstraction refinement based on spurious behaviors [HJRM04]
- Transition relation approximation [JM05]

- Bounded model checking: approximate reachability set computation [McM03]
- Predicate abstraction refinement based on spurious behaviors [HJRM04]
- Transition relation approximation [JM05]
- Lazy abstraction [McM06]

- Bounded model checking: approximate reachability set computation [McM03]
- Predicate abstraction refinement based on spurious behaviors [HJRM04]
- Transition relation approximation [JM05]
- Lazy abstraction [McM06]
- Software upgrade checking [Pincette,SFS12]

- Bounded model checking: approximate reachability set computation [McM03]
- Predicate abstraction refinement based on spurious behaviors [HJRM04]
- Transition relation approximation [JM05]
- Lazy abstraction [McM06]
- Software upgrade checking [Pincette,SFS12]

Property-based overapproximation

- Various applications, different interpolation requirements

- Various applications, different interpolation requirements
- Various interpolation systems [P97,McM04,DKPW10]

- Various applications, different interpolation requirements
- Various interpolation systems [P97,McM04,DKPW10]
- Various interpolant features

- Various applications, different interpolation requirements
- Various interpolation systems [P97,McM04,DKPW10]
- Various interpolant features
  - Strength affects overapproximation coarseness

- Various applications, different interpolation requirements
- Various interpolation systems [P97,McM04,DKPW10]
- Various interpolant features
  - Strength affects overapproximation coarseness
  - Strength empirically affects verification performance, convergence

Interpolation

Open Issues and Contribution

- Various applications, different interpolation requirements
- Various interpolation systems [P97,McM04,DKPW10]
- Various interpolant features
  - Strength affects overapproximation coarseness
  - Strength empirically affects verification performance, convergence

$\Rightarrow$ Formalization of requirements for simultaneous abstraction, path interpolation

- Various applications, different interpolation requirements
- Various interpolation systems [P97,McM04,DKPW10]
- Various interpolant features
    - Strength affects overapproximation coarseness
    - Strength empirically affects verification performance, convergence

⇒ Formalization of requirements for simultaneous abstraction, path interpolation

⇒ Identification of subset of interpolation systems satisfying requirements

Interpolation [Craig57,McM03]
Background

- Craig's interpolant I for unsatisfiable $A \wedge B$

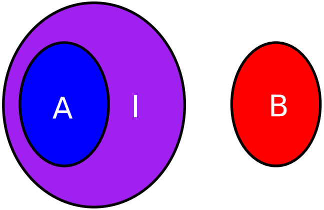
- Craig's interpolant I for unsatisfiable $A \wedge B$
  - $A \rightarrow I$ $\quad$ $I \wedge B$ unsatisfiable

- Craig's interpolant I for unsatisfiable $A \wedge B$
  - $A \rightarrow I$ $\quad$ $I \wedge B$ unsatisfiable
  - I defined over common symbols of A and B

- Craig's interpolant I for unsatisfiable $A \wedge B$
  - $A \rightarrow I$ $\quad$ $I \wedge B$ unsatisfiable
  - I defined over common symbols of A and B
  - I as overapproximation A conflicting with B

Interpolation [Craig57,McM03]
Background

- Craig's interpolant I for unsatisfiable $A \wedge B$
  - $A \rightarrow I$ $\qquad$ $I \wedge B$ unsatisfiable
  - I defined over common symbols of A and B
  - I as overapproximation A conflicting with B



Interpolant Strength

Applications to Symbolic Model Checking

- I_1 stronger than I_2 $\qquad I_1 \rightarrow I_2$

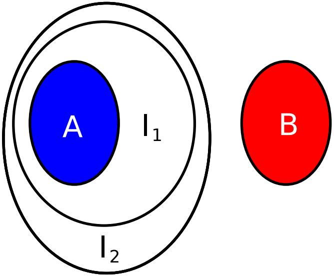
- I_1 stronger than I_2 $\qquad I_1 \rightarrow I_2$
- Interpolation as property-based overapproximation

Interpolant Strength

Applications to Symbolic Model Checking

- I_1 stronger than I_2 $\quad I_1 \rightarrow I_2$
- Interpolation as property-based overapproximation
- Strength affects approximation coarseness

- I_1 stronger than I_2 $\quad I_1 \rightarrow I_2$
- Interpolation as property-based overapproximation
- Strength affects approximation coarseness



- Interpolant I for unsatisfiable $A \wedge B$

- Interpolant I for unsatisfiable $A \wedge B$
- Different procedures [P97,McM04,DKPW10]

- Interpolant I for unsatisfiable $A \wedge B$
- Different procedures [P97,McM04,DKPW10]
- Standard generation approach

- Interpolant I for unsatisfiable $A \wedge B$
- Different procedures [P97,McM04,DKPW10]
- Standard generation approach
  - Derivation of unsatisfiability resolution proof of $A \wedge B$

- Interpolant I for unsatisfiable $A \wedge B$
- Different procedures [P97,McM04,DKPW10]
- Standard generation approach
  - Derivation of unsatisfiability resolution proof of $A \wedge B$
  - Computation of I from proof structure

- Propositional SAT $\qquad p \wedge (\overline{p} \vee r)$

- Propositional SAT $p \wedge (\overline{p} \vee r)$
- Proof of unsatisfiability

Proofs
SAT

- Propositional SAT $\quad p \wedge (\overline{p} \vee r)$
- Proof of unsatisfiability
  - Certificate of unsatisfiability

Proofs
SAT

- Propositional SAT $\qquad p \wedge (\overline{p} \vee r)$
- Proof of unsatisfiability
  - Certificate of unsatisfiability
  - Generated at solving time

Proofs
SAT

- Propositional SAT $p \wedge (\overline{p} \vee r)$
- Proof of unsatisfiability
  - Certificate of unsatisfiability
  - Generated at solving time
- CDCL SAT solver

- Propositional SAT $\quad p \wedge (\overline{p} \vee r)$
- Proof of unsatisfiability
    - Certificate of unsatisfiability
    - Generated at solving time
- CDCL SAT solver
    - Resolution system

Resolution System
Background

- Literal $p \quad \overline{p}$

Resolution System
Background

- Literal $p \quad \overline{p}$
- Clause $p \vee \overline{q} \vee r \vee \ldots \rightsquigarrow p\overline{q}r\ldots$ Empty clause $\perp$

Resolution System
Background

- Literal $\quad p \quad \overline{p}$
- Clause $\quad p \lor \overline{q} \lor r \lor \ldots \rightsquigarrow p\overline{q}r\ldots$ $\quad$ Empty clause $\quad \bot$
- Input formula $\quad (p \lor q) \land (r \lor \overline{p}) \land \ldots \rightsquigarrow \{pq, r\overline{p}, \ldots\}$

- Literal $p \quad \overline{p}$
- Clause $p \vee \overline{q} \vee r \vee \ldots \rightsquigarrow p\overline{q}r\ldots$ Empty clause $\perp$
- Input formula $(p \vee q) \wedge (r \vee \overline{p}) \wedge \ldots \rightsquigarrow \{pq, r\overline{p}, \ldots\}$
- Resolution rule

$$\frac{pC \qquad \overline{p}D}{CD}\, p$$

Antecedents: $pC\ \overline{p}D$ Resolvent: CD Pivot: p

Resolution Proofs

Resolution System

- Resolution proof of unsatisfiability of a set of clauses S

- Resolution proof of unsatisfiability of a set of clauses S
  - Tree
  - Leaves as clauses of S
  - Inner nodes as resolvents
  - Root as unique $\perp$

- Resolution proof of unsatisfiability of a set of clauses S
  - Tree
  - Leaves as clauses of S
  - Inner nodes as resolvents
  - Root as unique $\perp$
- Set of clauses $A = \{p\overline{q}, r\}$ $B = \{\overline{pr}, q\}$

- Resolution proof of unsatisfiability of a set of clauses S
  - Tree
  - Leaves as clauses of S
  - Inner nodes as resolvents
  - Root as unique $\perp$
- Set of clauses $A = \{p\overline{q}, r\}$ $B = \{\overline{pr}, q\}$
- Proof of unsatisfiability

$$
\cfrac{\cfrac{\cfrac{p\overline{q} \qquad \overline{pr}}{\overline{qr}}\,p \qquad r}{\overline{q}}\,r \qquad q}{\perp}\,q
$$

Outline

1. Background
   - Interpolation for Model Checking
   - Labeled Interpolation Systems
2. Contribution
   - Interpolant Strength in Model Checking
   - Simultaneous Abstraction: Requirements and Constraints
   - Path Interpolation: Requirements and Constraints
3. Conclusions

- Interpolation system parametric in labeling function [DKPW10]

- Interpolation system parametric in labeling function [DKPW10]

- Interpolant determined by proof and labeling

- Interpolation system parametric in labeling function [DKPW10]

- Interpolant determined by proof and labeling

- Generalization of [P97,McM04] (P, M, M')

- Interpolation system parametric in labeling function [DKPW10]
- Interpolant determined by proof and labeling
- Generalization of [P97,McM04] (P, M, M')
- Strength comparison can be reduced to labeling comparison

- Labeling L for $A \wedge B$

Labeling

Labeled Interpolation Systems

- Labeling L for $A \wedge B$
  - Label $\in \{a, b, ab\}$

- Labeling L for $A \wedge B$
  - Label $\in \{a, b, ab\}$
  - Individual clause literals

- Labeling L for $A \wedge B$
  - Label $\in \{a, b, ab\}$
  - Individual clause literals
- A-local $\mapsto a$, B-local $\mapsto b$, AB-common $\mapsto \{a, b, ab\}$

Labeling

Labeled Interpolation Systems

- Labeling L for $A \wedge B$
  - Label $\in \{a, b, ab\}$
  - Individual clause literals
- A-local $\mapsto a$, B-local $\mapsto b$, AB-common $\mapsto \{a, b, ab\}$
- $A = (\overset{a}{\overline{p}} \vee \overset{?}{\overline{q}}) \wedge (\overset{a}{p} \vee \overset{?}{\overline{q}})$ $\quad$ $B = (\overset{?}{q} \vee \overset{b}{\overline{r}}) \wedge (\overset{?}{q} \vee \overset{b}{r})$

Labeling Lattice [DKPW10]

Labeled Interpolation Systems

- $b \preceq ab \preceq a \qquad \rightsquigarrow \qquad (\alpha_1, \overset{L_1}{\ldots}, \alpha_n) \preceq (\beta_1, \overset{L_2}{\ldots}, \beta_n)$

Labeling Lattice [DKPW10]

Labeled Interpolation Systems

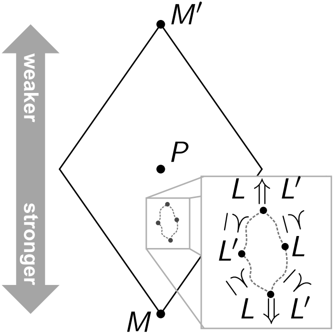
- $b \preceq ab \preceq a \quad \rightsquigarrow \quad (\alpha_1, \overset{L_1}{\ldots}, \alpha_n) \preceq (\beta_1, \overset{L_2}{\ldots}, \beta_n)$

- $(\alpha_1, \overset{L_1}{\ldots}, \alpha_n) \preceq (\beta_1, \overset{L_2}{\ldots}, \beta_n) \quad \Longrightarrow \quad I_1 \rightarrow I_2$

Labeling Lattice [DKPW10]

Labeled Interpolation Systems

- $b \preceq ab \preceq a \quad \rightsquigarrow \quad (\alpha_1, \overset{L_1}{\ldots}, \alpha_n) \preceq (\beta_1, \overset{L_2}{\ldots}, \beta_n)$
- $(\alpha_1, \overset{L_1}{\ldots}, \alpha_n) \preceq (\beta_1, \overset{L_2}{\ldots}, \beta_n) \quad \Longrightarrow \quad I_1 \to I_2$
- Labeling lattice



Outline

1 Background
- Interpolation for Model Checking
- Labeled Interpolation Systems

2 Contribution
- Interpolant Strength in Model Checking
- Simultaneous Abstraction: Requirements and Constraints
- Path Interpolation: Requirements and Constraints

3 Conclusions

Outline

1. Background
   - Interpolation for Model Checking
   - Labeled Interpolation Systems
2. Contribution
   - Interpolant Strength in Model Checking
   - Simultaneous Abstraction: Requirements and Constraints
   - Path Interpolation: Requirements and Constraints
3. Conclusions

- Systematic use of interpolation in symbolic model checking

- Systematic use of interpolation in symbolic model checking
- Focus on **interpolant strength**

- Systematic use of interpolation in symbolic model checking
- Focus on **interpolant strength**

$\Rightarrow$ **Scenarios**: simultaneous abstraction, path interpolation

- Systematic use of interpolation in symbolic model checking
- Focus on **interpolant strength**

$\Rightarrow$ **Scenarios**: simultaneous abstraction, path interpolation

  - Generation of multiple interpolants $I_1, \ldots, I_n$

- Systematic use of interpolation in symbolic model checking
- Focus on **interpolant strength**

⇒ **Scenarios**: simultaneous abstraction, path interpolation

  - Generation of multiple interpolants $I_1, \dots, I_n$
  - Additional requirements on $I_1, \dots, I_n$

- Systematic use of interpolation in symbolic model checking

- Focus on **interpolant strength**

⇒ **Scenarios**: simultaneous abstraction, path interpolation

  - Generation of multiple interpolants $I_1, \ldots, I_n$
  - Additional requirements on $I_1, \ldots, I_n$

⇒ **Constraints on labeled interpolation systems**

- Systematic use of interpolation in symbolic model checking
- Focus on **interpolant strength**

$\Rightarrow$ **Scenarios**: simultaneous abstraction, path interpolation

- Generation of multiple interpolants $I_1, \ldots, I_n$
- Additional requirements on $I_1, \ldots, I_n$

$\Rightarrow$ **Constraints on labeled interpolation systems**

- Generation of each I_i with different L_i

- Systematic use of interpolation in symbolic model checking
- Focus on **interpolant strength**

⇒ **Scenarios**: simultaneous abstraction, path interpolation

  - Generation of multiple interpolants $I_1, \ldots, I_n$
  - Additional requirements on $I_1, \ldots, I_n$

⇒ **Constraints on labeled interpolation systems**

  - Generation of each I_i with different L_i
  - Identification of constraints on $L_1, \ldots, L_n$

- Simultaneous abstraction

- Simultaneous abstraction
  - Software upgrade checking [Pincette,SFS12]

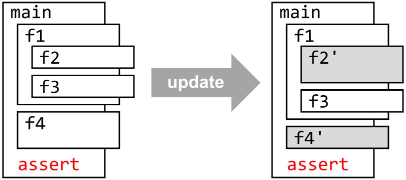
- Simultaneous abstraction
  - Software upgrade checking [Pincette,SFS12]
- Path interpolation

- Simultaneous abstraction
    - Software upgrade checking [Pincette,SFS12]
- Path interpolation
    - Counterexample-guided abstraction refinement [CGJLV00]

Outline

1. Background
   - Interpolation for Model Checking
   - Labeled Interpolation Systems
2. Contribution
   - Interpolant Strength in Model Checking
   - Simultaneous Abstraction: Requirements and Constraints
   - Path Interpolation: Requirements and Constraints
3. Conclusions

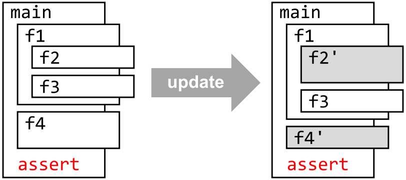
Software Upgrade Checking
Simultaneous Abstraction



- Program safe

Software Upgrade Checking

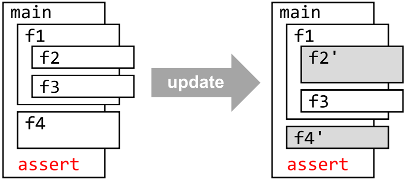
Simultaneous Abstraction



- Program safe $\phi_{main} \wedge \phi_{f1} \wedge \phi_{f2} \wedge \phi_{f3} \wedge \phi_{f4}$ UNSAT

Software Upgrade Checking

Simultaneous Abstraction

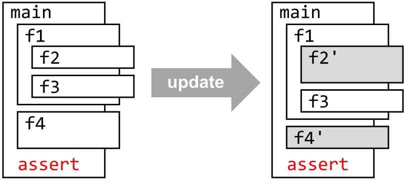


- Program safe $\phi_{main} \wedge \phi_{f1} \wedge \phi_{f2} \wedge \phi_{f3} \wedge \phi_{f4}$ UNSAT

- Extract interpolants

Software Upgrade Checking

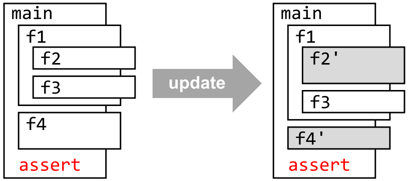
Simultaneous Abstraction



- Program safe $\phi_{main} \wedge \phi_{f1} \wedge \phi_{f2} \wedge \phi_{f3} \wedge \phi_{f4}$ UNSAT
- Extract interpolants $I_{main} \wedge I_{f1} \wedge I_{f2} \wedge I_{f3} \wedge I_{f4}$ UNSAT

Software Upgrade Checking

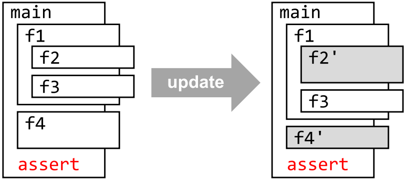
Simultaneous Abstraction



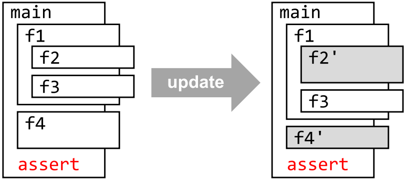
- Program safe $\phi_{main} \wedge \phi_{f1} \wedge \phi_{f2} \wedge \phi_{f3} \wedge \phi_{f4}$ UNSAT
- Extract interpolants $I_{main} \wedge I_{f1} \wedge I_{f2} \wedge I_{f3} \wedge I_{f4}$ UNSAT
- Functions update $f_2 \rightsquigarrow f_2'$ $f_4 \rightsquigarrow f_4'$

Software Upgrade Checking

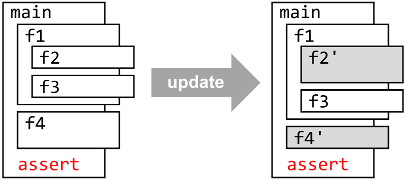
Simultaneous Abstraction



- Program safe $\phi_{main} \wedge \phi_{f1} \wedge \phi_{f2} \wedge \phi_{f3} \wedge \phi_{f4}$ UNSAT
- Extract interpolants $I_{main} \wedge I_{f1} \wedge I_{f2} \wedge I_{f3} \wedge I_{f4}$ UNSAT
- Functions update $f_2 \rightsquigarrow f_2'$ $f_4 \rightsquigarrow f_4'$
- Program safe?



- Program safe $\phi_{main} \wedge \phi_{f1} \wedge \phi_{f2} \wedge \phi_{f3} \wedge \phi_{f4}$ UNSAT
- Extract interpolants $I_{main} \wedge I_{f1} \wedge I_{f2} \wedge I_{f3} \wedge I_{f4}$ UNSAT
- Functions update $f_2 \rightsquigarrow f_2'$ $f_4 \rightsquigarrow f_4'$
- Program safe? Check $\phi_{f'2} \rightarrow I_{f2}$ $\phi_{f'4} \rightarrow I_{f4}$



- Program safe $\phi_{main} \wedge \phi_{f1} \wedge \phi_{f2} \wedge \phi_{f3} \wedge \phi_{f4}$ UNSAT
- Extract interpolants $I_{main} \wedge I_{f1} \wedge I_{f2} \wedge I_{f3} \wedge I_{f4}$ UNSAT
- Functions update $f_2 \rightsquigarrow f_2'$ $f_4 \rightsquigarrow f_4'$
- Program safe? Check $\phi_{f'2} \rightarrow I_{f2}$ $\phi_{f'4} \rightarrow I_{f4}$

- Requirement: $I_1 \wedge \ldots \wedge I_n$ UNSAT

Results

Simultaneous Abstraction

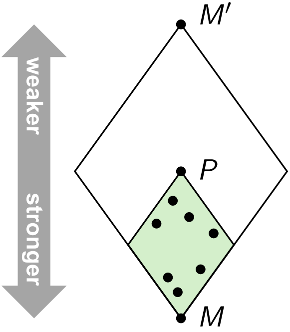
- Requirement: $I_1 \wedge \ldots \wedge I_n$ UNSAT
- Satisfied for: $L_1, \ldots, L_n \preceq$ Pudlák

Results

Simultaneous Abstraction

- Requirement: $I_1 \wedge \ldots \wedge I_n$ UNSAT
- Satisfied for: $L_1, \ldots, L_n \preceq$ Pudlák
- Not satisfied in general for: $L_i \succ$ Pudlák

- Requirement: $I_1 \wedge \ldots \wedge I_n$ UNSAT
- Satisfied for: $L_1, \ldots, L_n \preceq$ Pudlák
- Not satisfied in general for: $L_i \succ$ Pudlák



Outline

1. Background
   - Interpolation for Model Checking
   - Labeled Interpolation Systems
2. Contribution
   - Interpolant Strength in Model Checking
   - Simultaneous Abstraction: Requirements and Constraints
   - Path Interpolation: Requirements and Constraints
3. Conclusions

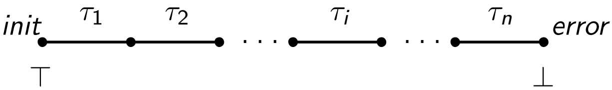
CEGAR
Path Interpolation

- Counterexample-guided abstraction refinement
  - Abstract $\rightarrow$ Check $\rightarrow$ Refine

CEGAR
Path Interpolation

- Counterexample-guided abstraction refinement
  - Abstract $\rightarrow$ Check $\rightarrow$ Refine
- Spurious trace

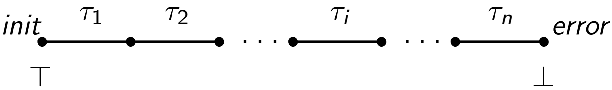
- Counterexample-guided abstraction refinement
  - Abstract $\rightarrow$ Check $\rightarrow$ Refine
- Spurious trace $\tau_1 \wedge \ldots \wedge \tau_n$ UNSAT



CEGAR

Path Interpolation

- Counterexample-guided abstraction refinement
  - Abstract $\rightarrow$ Check $\rightarrow$ Refine
- Spurious trace $\tau_1 \wedge \ldots \wedge \tau_n$ UNSAT

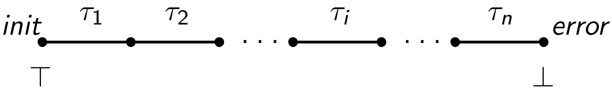


- Extract interpolants

CEGAR

Path Interpolation

- Counterexample-guided abstraction refinement
  - Abstract $\rightarrow$ Check $\rightarrow$ Refine
- Spurious trace $\tau_1 \wedge \ldots \wedge \tau_n$ UNSAT



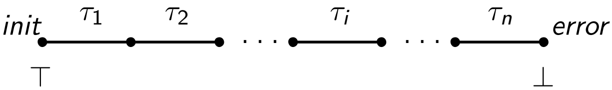
- Extract interpolants

  $\top \wedge \tau_1 \rightarrow I_1 \qquad I_i \wedge \tau_{i+1} \rightarrow I_{i+1} \qquad I_{n-1} \wedge \tau_n \rightarrow \bot$

CEGAR

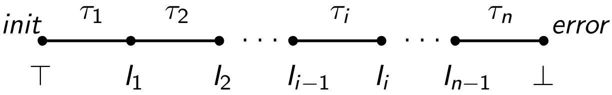
Path Interpolation

- Counterexample-guided abstraction refinement
  - Abstract $\rightarrow$ Check $\rightarrow$ Refine
- Spurious trace $\quad \tau_1 \wedge \ldots \wedge \tau_n \quad$ UNSAT



- Extract interpolants

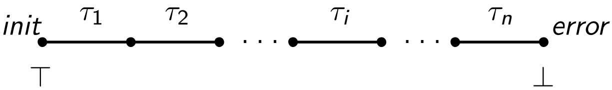
$\top \wedge \tau_1 \rightarrow I_1 \qquad I_i \wedge \tau_{i+1} \rightarrow I_{i+1} \qquad I_{n-1} \wedge \tau_n \rightarrow \bot$



CEGAR

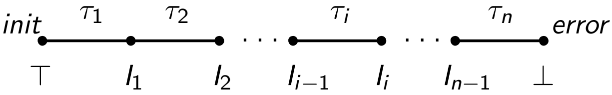
Path Interpolation

- Counterexample-guided abstraction refinement
  - Abstract $\rightarrow$ Check $\rightarrow$ Refine
- Spurious trace $\tau_1 \wedge \ldots \wedge \tau_n$ UNSAT



- Extract interpolants

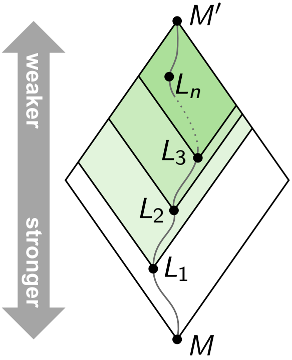
$\top \wedge \tau_1 \rightarrow I_1$ $\quad I_i \wedge \tau_{i+1} \rightarrow I_{i+1}$ $\quad I_{n-1} \wedge \tau_n \rightarrow \bot$



- Requirement: $\tau_1 \to I_1 \qquad I_i \wedge \tau_{i+1} \to I_{i+1} \qquad I_{n-1} \wedge \tau_n \to \bot$

- Requirement: $\tau_1 \to I_1 \qquad I_i \wedge \tau_{i+1} \to I_{i+1} \qquad I_{n-1} \wedge \tau_n \to \bot$
- Satisfied for: $L_1 \preceq \ldots \preceq L_n$

- Requirement: $\tau_1 \rightarrow I_1 \qquad I_i \wedge \tau_{i+1} \rightarrow I_{i+1} \qquad I_{n-1} \wedge \tau_n \rightarrow \perp$
- Satisfied for: $L_1 \preceq \ldots \preceq L_n$



Outline

1. Background
   - Interpolation for Model Checking
   - Labeled Interpolation Systems
2. Contribution
   - Interpolant Strength in Model Checking
   - Simultaneous Abstraction: Requirements and Constraints
   - Path Interpolation: Requirements and Constraints
3. Conclusions

Summary

- Interpolant strength in symbolic model checking

Summary

- Interpolant strength in symbolic model checking
- Simultaneous abstraction, path interpolation

Summary

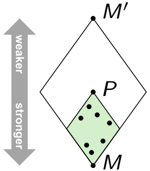
- Interpolant strength in symbolic model checking
- Simultaneous abstraction, path interpolation
  - Generation of multiple interpolants, additional requirements

Summary

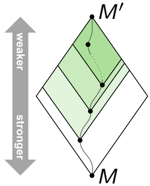
- Interpolant strength in symbolic model checking
- Simultaneous abstraction, path interpolation
  - Generation of multiple interpolants, additional requirements
- Constraints on labeled interpolation systems

Summary

- Interpolant strength in symbolic model checking
- Simultaneous abstraction, path interpolation
  - Generation of multiple interpolants, additional requirements
- Constraints on labeled interpolation systems



Sim. Abstraction



Path Interpolation

Ongoing Work

- Necessary and sufficient conditions for labeled interpolation systems

Ongoing Work

- Necessary and sufficient conditions for labeled interpolation systems
  - Path interpolation, simultaneous abstraction, state-transition interpolation, tree interpolation

Ongoing Work

- Necessary and sufficient conditions for labeled interpolation systems
    - Path interpolation, simultaneous abstraction, state-transition interpolation, tree interpolation
- Labeled interpolation systems w.r.t. semantical/syntactical features of interpolants

Ongoing Work

- Necessary and sufficient conditions for labeled interpolation systems
  - Path interpolation, simultaneous abstraction, state-transition interpolation, tree interpolation
- Labeled interpolation systems w.r.t. semantical/syntactical features of interpolants
  - Experimentation with FunFrog, eVolCheck, SAFARI model checkers

Ongoing Work

- Necessary and sufficient conditions for labeled interpolation systems
  - Path interpolation, simultaneous abstraction, state-transition interpolation, tree interpolation
- Labeled interpolation systems w.r.t. semantical/syntactical features of interpolants
  - Experimentation with FunFrog, eVolCheck, SAFARI model checkers

Thanks for your attention!

`verify.inf.usi.ch`

Reference

S.F. Rollini, O. Sery and N. Sharygina
Leveraging Interpolant Strength in Model Checking.
CAV 2012.

Example

Labeled Interpolation Systems

- $A = \{p\overline{q}, r\} \qquad B = \{\overline{pr}, q\}$

- $A = \{p\overline{q}, r\}$ $\qquad$ $B = \{\overline{pr}, q\}$

$p, q, r \mapsto (b, b, b)$

$$\dfrac{\dfrac{\dfrac{p\overline{q} \qquad \overline{pr}}{\overline{qr}} \qquad r}{\overline{q}} \qquad q}{\bot\,[(p \vee \overline{q}) \wedge r]}$$

Example

Labeled Interpolation Systems

- $A = \{p\overline{q}, r\}$ $\qquad$ $B = \{\overline{pr}, q\}$

$$p, q, r \mapsto (b, b, b)$$

$$\dfrac{\dfrac{\dfrac{p\overline{q} \quad \overline{pr}}{\overline{qr}} \quad r}{\overline{q}} \quad q}{\perp [(p \vee \overline{q}) \wedge r]}$$

$$p, q, r \mapsto (a, a, a)$$

$$\dfrac{\dfrac{\dfrac{p\overline{q} \quad \overline{pr}}{\overline{qr}} \quad r}{\overline{q}} \quad q}{\perp [(p \wedge r) \vee \overline{q}]}$$

- $A = \{p\overline{q}, r\}$ $\qquad B = \{\overline{pr}, q\}$

$$p, q, r \mapsto (b, b, b)$$

$$\dfrac{\dfrac{\dfrac{p\overline{q} \quad \overline{pr}}{\overline{qr}} \qquad r}{\overline{q}} \qquad q}{\bot\ [(p \vee \overline{q}) \wedge r]}$$

$$p, q, r \mapsto (a, a, a)$$

$$\dfrac{\dfrac{\dfrac{p\overline{q} \quad \overline{pr}}{\overline{qr}} \qquad r}{\overline{q}} \qquad q}{\bot\ [(p \wedge r) \vee \overline{q}]}$$

- $(b, b, b) \preceq (a, a, a)$

Example

Labeled Interpolation Systems

- $A = \{p\overline{q}, r\}$ $\qquad$ $B = \{\overline{pr}, q\}$

$$p, q, r \mapsto (b, b, b) \qquad\qquad p, q, r \mapsto (a, a, a)$$

$$\dfrac{\dfrac{\dfrac{p\overline{q} \quad \overline{pr}}{\overline{qr}} \quad r}{\overline{q}} \quad q}{\bot\,[(p \vee \overline{q}) \wedge r]} \qquad\qquad \dfrac{\dfrac{\dfrac{p\overline{q} \quad \overline{pr}}{\overline{qr}} \quad r}{\overline{q}} \quad q}{\bot\,[(p \wedge r) \vee \overline{q}]}$$

- $(b, b, b) \preceq (a, a, a) \quad \Longrightarrow \quad (p \vee \overline{q}) \wedge r \rightarrow (p \wedge r) \vee \overline{q}$

Transition Relation Approximation

Simultaneous Abstraction

- BMC: iterative analysis k-length traces

- BMC: iterative analysis k-length traces
  - Initial states S, abstract transition relation $\hat{T}$, error states E

Transition Relation Approximation

Simultaneous Abstraction

- BMC: iterative analysis k-length traces
  - Initial states S, abstract transition relation $\hat{T}$, error states E
- Check $$S \wedge \hat{T}^0 \wedge \hat{T}^1 \wedge \ldots \wedge \hat{T}^{k-1} \wedge E^k$$

- BMC: iterative analysis k-length traces
  - Initial states S, abstract transition relation $\hat{T}$, error states E
- Check $S \wedge \hat{T}^0 \wedge \hat{T}^1 \wedge \ldots \wedge \hat{T}^{k-1} \wedge E^k$ SAT?

- BMC: iterative analysis k-length traces
    - Initial states S, abstract transition relation $\hat{T}$, error states E
- Check $S \wedge \hat{T}^0 \wedge \hat{T}^1 \wedge \ldots \wedge \hat{T}^{k-1} \wedge E^k$ SAT?
- Check $S \wedge T^0 \wedge T^1 \wedge \ldots \wedge T^{k-1} \wedge E^k$

- BMC: iterative analysis k-length traces
    - Initial states S, abstract transition relation $\hat{T}$, error states E

- Check $S \wedge \hat{T}^0 \wedge \hat{T}^1 \wedge \ldots \wedge \hat{T}^{k-1} \wedge E^k$ SAT?

- Check $S \wedge T^0 \wedge T^1 \wedge \ldots \wedge T^{k-1} \wedge E^k$ UNSAT?

Transition Relation Approximation
Simultaneous Abstraction

- BMC: iterative analysis k-length traces
  - Initial states S, abstract transition relation $\hat{T}$, error states E
- Check $S \wedge \hat{T}^0 \wedge \hat{T}^1 \wedge \ldots \wedge \hat{T}^{k-1} \wedge E^k$ SAT?
- Check $S \wedge T^0 \wedge T^1 \wedge \ldots \wedge T^{k-1} \wedge E^k$ UNSAT?
- Extract interpolants

Transition Relation Approximation

Simultaneous Abstraction

- BMC: iterative analysis k-length traces
  - Initial states S, abstract transition relation $\hat{T}$, error states E
- Check $S \wedge \hat{T}^0 \wedge \hat{T}^1 \wedge \ldots \wedge \hat{T}^{k-1} \wedge E^k$ SAT?
- Check $S \wedge T^0 \wedge T^1 \wedge \ldots \wedge T^{k-1} \wedge E^k$ UNSAT?
- Extract interpolants $S \wedge I_0 \wedge I_1 \wedge \ldots \wedge I_{k-1} \wedge E^k$ UNSAT

- BMC: iterative analysis k-length traces
  - Initial states S, abstract transition relation $\hat{T}$, error states E
- Check $S \wedge \hat{T}^0 \wedge \hat{T}^1 \wedge \ldots \wedge \hat{T}^{k-1} \wedge E^k$ SAT?
- Check $S \wedge T^0 \wedge T^1 \wedge \ldots \wedge T^{k-1} \wedge E^k$ UNSAT?
- Extract interpolants $S \wedge I_0 \wedge I_1 \wedge \ldots \wedge I_{k-1} \wedge E^k$ UNSAT
- Strengthen $\hat{T}$

Transition Relation Approximation
Simultaneous Abstraction

- BMC: iterative analysis k-length traces
  - Initial states S, abstract transition relation $\hat{T}$, error states E

- Check $S \wedge \hat{T}^0 \wedge \hat{T}^1 \wedge \ldots \wedge \hat{T}^{k-1} \wedge E^k$ SAT?
- Check $S \wedge T^0 \wedge T^1 \wedge \ldots \wedge T^{k-1} \wedge E^k$ UNSAT?
- Extract interpolants $S \wedge I_0 \wedge I_1 \wedge \ldots \wedge I_{k-1} \wedge E^k$ UNSAT
- Strengthen $\hat{T}$ $\hat{T} \wedge I_0 \wedge I_1 \wedge \ldots \wedge I_{k-1} \quad \rightsquigarrow \quad \hat{\hat{T}}$

- BMC: iterative analysis k-length traces
  - Initial states S, abstract transition relation $\hat{T}$, error states E

- Check $S \wedge \hat{T}^0 \wedge \hat{T}^1 \wedge \ldots \wedge \hat{T}^{k-1} \wedge E^k$ SAT?
- Check $S \wedge T^0 \wedge T^1 \wedge \ldots \wedge T^{k-1} \wedge E^k$ UNSAT?
- Extract interpolants $S \wedge I_0 \wedge I_1 \wedge \ldots \wedge I_{k-1} \wedge E^k$ UNSAT
- Strengthen $\hat{T}$ $\hat{T} \wedge I_0 \wedge I_1 \wedge \ldots \wedge I_{k-1} \quad \rightsquigarrow \quad \hat{\hat{T}}$
- $S \wedge \hat{\hat{T}}^0 \wedge \hat{\hat{T}}^1 \wedge \ldots \wedge \hat{\hat{T}}^{k-1} \wedge E^k$ UNSAT!

Transition Relation Approximation

Simultaneous Abstraction

- BMC: iterative analysis k-length traces
  - Initial states S, abstract transition relation $\hat{T}$, error states E

- Check $\quad S \wedge \hat{T}^0 \wedge \hat{T}^1 \wedge \ldots \wedge \hat{T}^{k-1} \wedge E^k \quad$ SAT?
- Check $\quad S \wedge T^0 \wedge T^1 \wedge \ldots \wedge T^{k-1} \wedge E^k \quad$ UNSAT?
- Extract interpolants $\quad S \wedge I_0 \wedge I_1 \wedge \ldots \wedge I_{k-1} \wedge E^k \quad$ UNSAT
- Strengthen $\hat{T}$ $\quad \hat{T} \wedge I_0 \wedge I_1 \wedge \ldots \wedge I_{k-1} \quad \rightsquigarrow \quad \hat{\hat{T}}$
- $\quad S \wedge \hat{\hat{T}}^0 \wedge \hat{\hat{T}}^1 \wedge \ldots \wedge \hat{\hat{T}}^{k-1} \wedge E^k \quad$ UNSAT!