Refinement of Trace Abstraction

Tuesday, December 15, 2011

Craig interpolants

Craig interpolant - logical formulas

Given: Unsatisfiable conjuction $\boxed{\mathtt{A}} \wedge \boxed{\mathtt{B}}$

Interpolant is a formula $\boxed{\mathtt{I}}$ such that:

- $\boxed{\mathtt{A}}$ implies $\boxed{\mathtt{I}}$ and $\boxed{\mathtt{I}} \wedge \boxed{\mathtt{B}}$ unsatisfiable
- $\boxed{\mathtt{I}}$ contains only common symbols of A and B

William Craig

Three uses of the Herbrand-Gentzen theorem in relating model theory and proof theory (

Journal of Sybolic Logic (1957))

Craig interpolants

Craig interpolant - logical formulas

Given: Unsatisfiable conjuction $\boxed{\texttt{A}} \wedge \boxed{\texttt{B}}$

Interpolant is a formula $\boxed{\texttt{I}}$ such that:

- $\boxed{\texttt{A}}$ implies $\boxed{\texttt{I}}$ and $\boxed{\texttt{I}} \wedge \boxed{\texttt{B}}$ unsatisfiable
- $\boxed{\texttt{I}}$ contains only common symbols of A and B

Example (propositional logic)

unsatisfiable conjuction: $\boxed{p \wedge q} \quad \wedge \quad \boxed{\neg p \wedge r}$

possible Craig interpolant: $\boxed{p}$

Example (SMT)

unsatisfiable conjuction: $\boxed{f(x_1) = y \wedge x_1 = x_2} \quad \wedge \quad \boxed{x_2 = x_3 \wedge f(x_3) \neq y}$

possible Craig interpolant: $\boxed{y = f(x_2)}$

Interpolants

Interpolant - execution traces

Given: Infeasible trace $st_1 \dots st_i\, st_{i+1} \dots st_n$

Interpolant is assertion $\boxed{\texttt{I}}$ such that:

- $post(\boxed{\texttt{true}}, st_1 \dots st_i) \subseteq \boxed{\texttt{I}} \subseteq wp(\boxed{\texttt{false}}, st_{i+1} \dots st_n)$
- $\boxed{\texttt{I}}$ contains only program variables occuring in both, $st_1 ... st_i$ and $st_{i+1} ... st_n$

Kenneth L. McMillan

Interpolation and SAT-Based Model Checking (CAV 2003)

Interpolants

Interpolant - execution traces

Given: Infeasible trace $st_1 \ldots st_i \, st_{i+1} \ldots st_n$

Interpolant is assertion $\boxed{\mathtt{I}}$ such that:

- $post(\boxed{\mathtt{true}}, st_1 \ldots st_i) \subseteq \boxed{\mathtt{I}} \subseteq wp(\boxed{\mathtt{false}}, st_{i+1} \ldots st_n)$
- $\boxed{\mathtt{I}}$ contains only program variables occuring in both, $st_1 ... st_i$ and $st_{i+1} ... st_n$

Example

infeasible trace: `x:=0` `y:=0` `x++` `x==-1`

possible interpolant: $\boxed{x \geq 0}$

Inductive interpolants

Inductive sequence of interpolants

Given: Infeasible trace $st_1 \ldots st_n$

There exists sequence of assertions $I_0 \ldots I_n$ such that:

- $post(I_i, st_i) \subseteq I_{i+1}$
- $I_0 = \texttt{true}$ and $I_n = \texttt{false}$
- I_i contains only variables occuring in both, $st_1 \ldots st_i$ and $st_{i+1} \ldots st_n$

Ranjit Jhala, Kenneth L. McMillan

A Practical and Complete Approach to Predicate Refinement (TACAS 2006)

Inductive interpolants

Inductive sequence of interpolants

Given: Infeasible trace $st_1 \ldots st_n$

There exists sequence of assertions $I_0 \ldots I_n$ such that:

- $post(I_i, st_i) \subseteq I_{i+1}$
- $I_0 =$ `true` and $I_n =$ `false`
- I_i contains only variables occuring in both, $st_1 \ldots st_i$ and $st_{i+1} \ldots st_n$

Example

`true` —`x:=0`→ $x \geq 0$ —`y:=0`→ $x \geq 0$ —`x++`→ $x \geq 0$ —`x==-1`→ `false`

Computation of interpolants - Example

infeasible trace

`x:=0` `y:=0` `x++` `x==-1`

Computation of interpolants - Example

infeasible trace

`x:=0` `y:=0` `x++` `x==-1`

single static assignment form

$$x_0 = 0 \quad \wedge \quad y_1 = 0 \quad \wedge \quad x_2 = x_0 + 1 \quad \wedge \quad x_2 = -1$$

Computation of interpolants - Example

infeasible trace

`x:=0` `y:=0` `x++` `x==-1`

single static assignment form

$$x_0 = 0 \quad \wedge \quad y_1 = 0 \quad \wedge \quad x_2 = x_0 + 1 \quad \wedge \quad x_2 = -1$$

inductive sequence of Craig interpolants

`true`

Computation of interpolants - Example

infeasible trace

`x:=0` `y:=0` `x++` `x==-1`

single static assignment form

$x_0 = 0 \wedge y_1 = 0 \wedge x_2 = x_0 + 1 \wedge x_2 = -1$

inductive sequence of Craig interpolants

`true` $x_0 \geq 0$

Computation of interpolants - Example

infeasible trace

`x:=0` `y:=0` `x++` `x==-1`

single static assignment f

$x_0 = 0 \quad \wedge \quad y_1 = 0 \quad \wedge \quad \boxed{x_2 = x_0 + 1 \quad \wedge \quad x_2 = -1}$

inductive seque interpolants

$\boxed{\texttt{true}}$ $\boxed{x_0 \geq 0}$ $\boxed{x_0 \geq 0}$

Computation of interpolants - Example

infeasible trace

`x:=0` `y:=0` `x++` `x==-1`

single static assignment form

$x_0 = 0 \quad \wedge \quad y_1 = 0 \quad \wedge \quad x_2 = x_0 + 1 \quad \wedge \quad \boxed{x_2 = -1}$

inductive sequence of Craig int

$\boxed{\texttt{true}}$ $\boxed{x_0 \geq 0}$ $\boxed{x_0 \geq 0}$ $\boxed{x_2 \geq 0}$

Computation of interpolants - Example

infeasible trace

`x:=0` `y:=0` `x++` `x==-1`

single static assignment form

$$x_0 = 0 \quad \wedge \quad y_1 = 0 \quad \wedge \quad x_2 = x_0 + 1 \quad \wedge \quad x_2 = -1$$

inductive sequence of Craig interpolants

`true` $\quad x_0 \geq 0 \quad x_0 \geq 0 \quad x_2 \geq 0 \quad$ `false`

Computation of interpolants - Example

infeasible trace

`x:=0` `y:=0` `x++` `x==-1`

single static assignment form

$$x_0 = 0 \quad \wedge \quad y_1 = 0 \quad \wedge \quad x_2 = x_0 + 1 \quad \wedge \quad x_2 = -1$$

inductive sequence of Craig interpolants

$\boxed{\texttt{true}}$ $\boxed{x_0 \geq 0}$ $\boxed{x_0 \geq 0}$ $\boxed{x_2 \geq 0}$ $\boxed{\texttt{false}}$

inductive sequence of interpolants

$\boxed{\texttt{true}}$ $\boxed{x \geq 0}$ $\boxed{x \geq 0}$ $\boxed{x \geq 0}$ $\boxed{\texttt{false}}$

SmtInterpol

- SMT-Solver
  Computes sequences of Craig interpolants for the quantifier free combined theory of uninterpreted functions and linear arithmetic over rationals and integers.

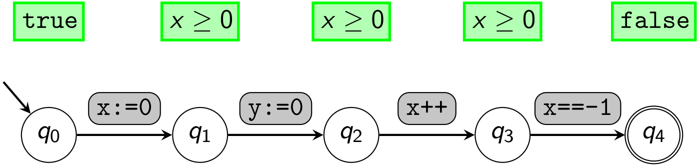
- Developed by

Jürgen Christ

Jochen Hoenicke

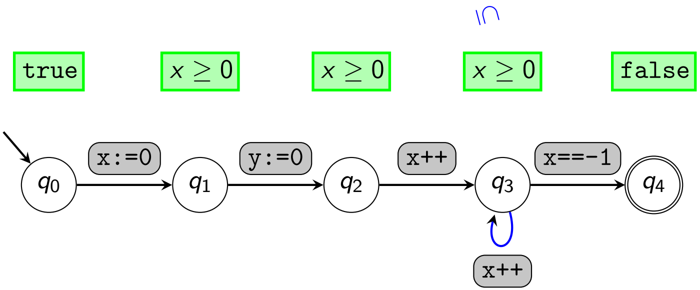
- http://swt.informatik.uni-freiburg.de/research/tools/smtinterpol

Example – Use Interpolants to Generalize Infeasible Traces



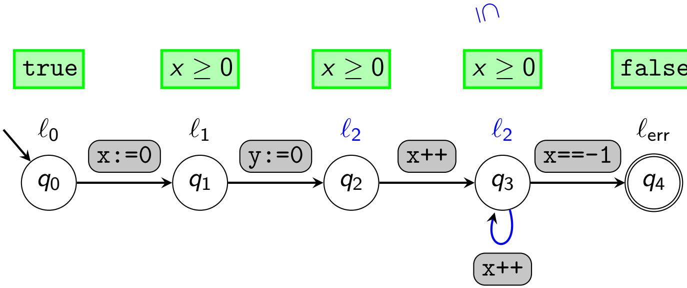
Example – Use Interpolants to Generalize Infeasible Traces

$$\text{post}(\boxed{x \geq 0}, \boxed{\texttt{x++}}) = x \geq 1$$

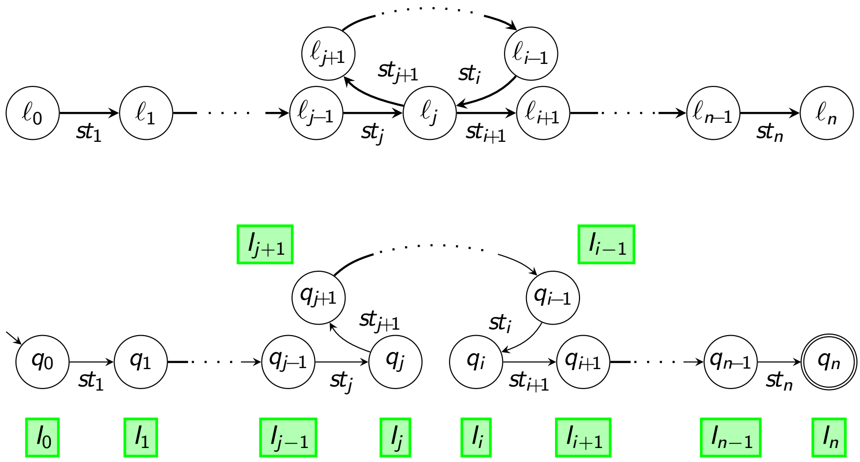


Example – Use Interpolants to Generalize Infeasible Traces

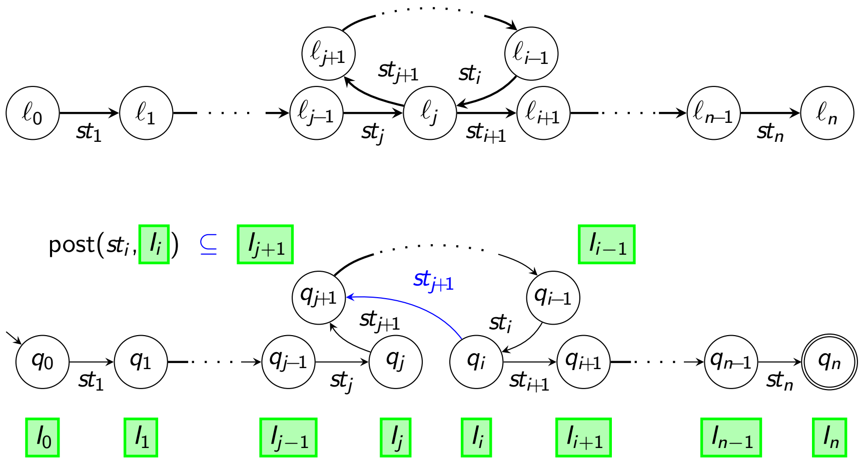
$$\text{post}(\boxed{x \geq 0}, \boxed{\texttt{x++}}) = x \geq 1$$



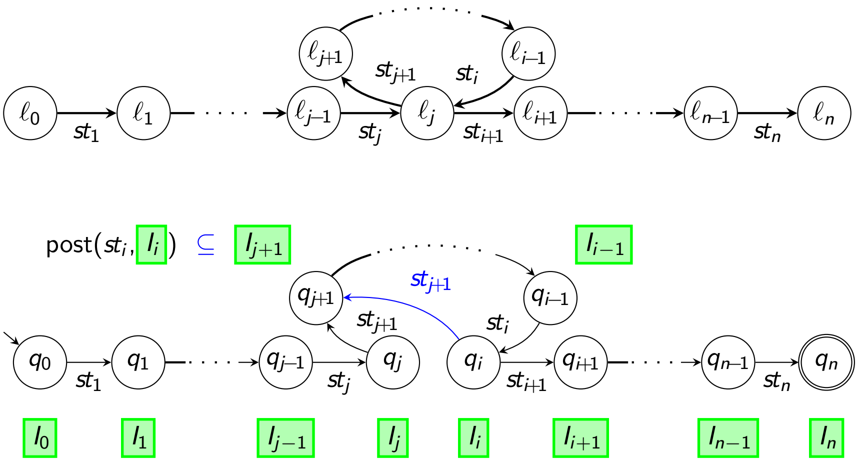
Schematic Example – Use Interpolants for Generalization



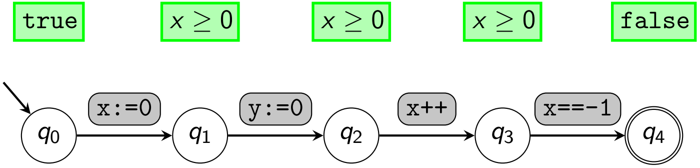
Schematic Example – Use Interpolants for Generalization



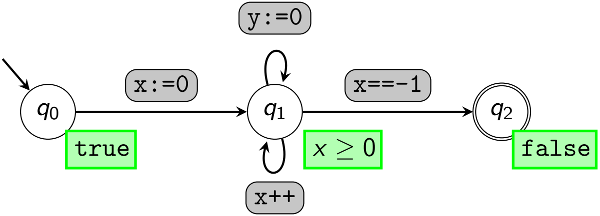
Schematic Example – Use Interpolants for Generalization



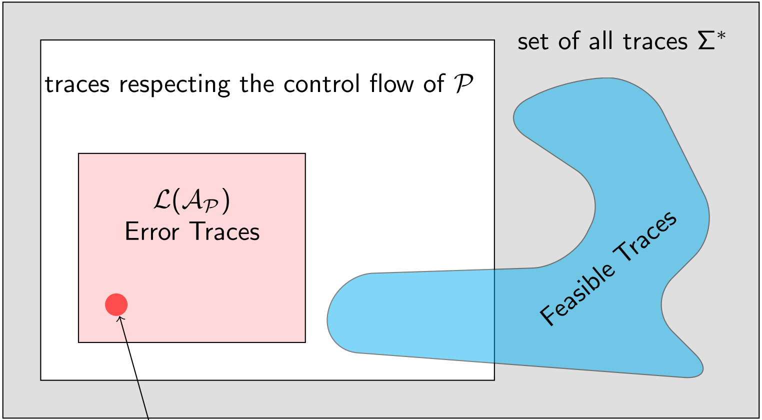
Example – Use Interpolants to Generalize Infeasible Traces



Interpolant automaton
obtained by merging all states labelled with same interpolant

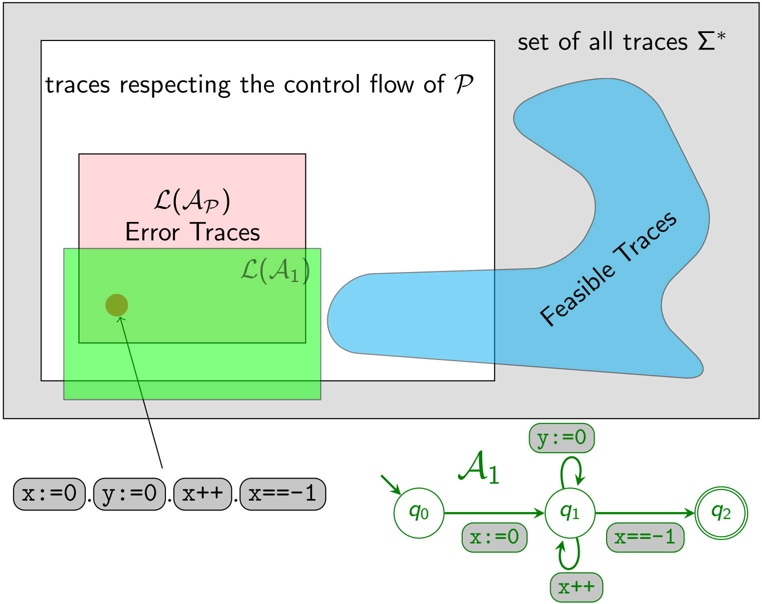


Example – Refinement Using Interpolant Automata

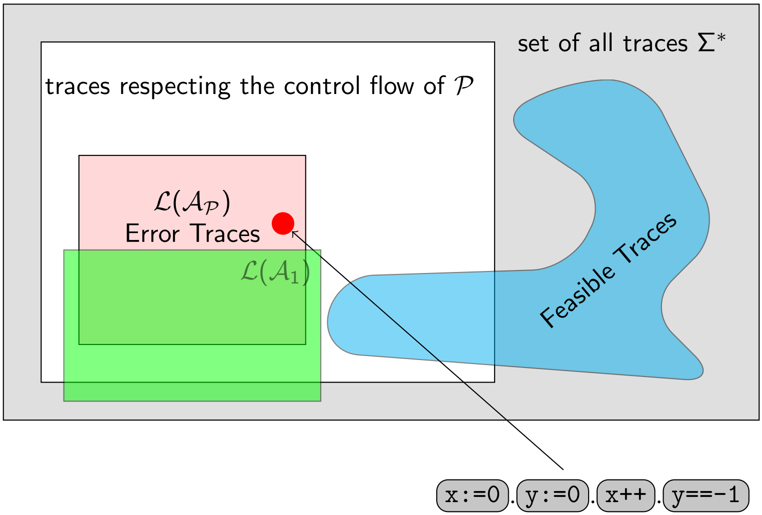


`x:=0`.`y:=0`.`x++`.`x==-1`

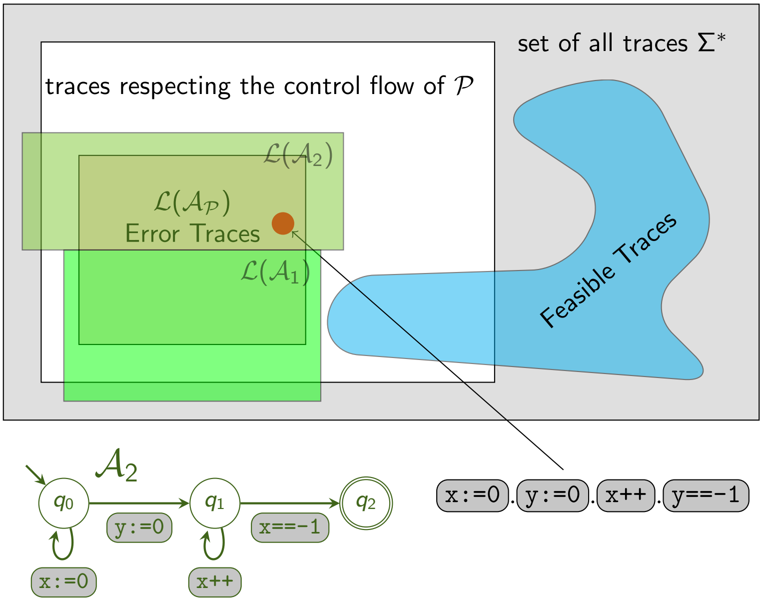
Example – Refinement Using Interpolant Automata



Example – Refinement Using Interpolant Automata



Example – Refinement Using Interpolant Automata



CEGAR for Trace Abstraction