On interpolation in theorem proving

Maria Paola Bonacina

Visiting: Computer Science Laboratory, SRI International, Menlo Park, CA, USA
Affiliation: Dipartimento di Informatica,
Università degli Studi di Verona, Verona, Italy, EU

Invited talk at the 7th Summer School on Formal Techniques (SSFT)
SRI International and Menlo College
Atherton, California, USA

May 26, 2017

What is interpolation?

- Consider a function f (univariate for simplicity)
- We know the values of f at points $x_1, \ldots, x_n$ on the x-axis (e.g., from sampling or experiments)
- We want to know the values of f at additional intermediate points and build its curve
- This is the problem of interpolation in numerical analysis
- It has many applications in computer graphics (e.g., spline interpolation)

Interpolation in logic

What is interpolation in logic?

Signature

- A finite set of constant symbols: e.g., a, b, c ...
- A finite set of function symbols: e.g., f, g, h ...
- A finite set of predicate symbols: P, Q, R, $\simeq$ ...
- Arities
- Sorts (important but key concepts can be understood without)

An infinite supply of variables: x, y, z, w ...

Logical language

- Terms: a, x, $f(a)$, $f(x)$, $g(a,x)$ ...
- Atoms: R, $P(a)$, $Q(x, g(b))$, ...
- Literals: R, $P(a)$, $Q(x, g(b))$, $\neg R$, $\neg P(a)$, $\neg Q(x, g(b))$, ...
- Formulae: $P(a) \wedge Q(a, g(b))$, $\neg P(a) \vee Q(a, g(b))$, $\neg P(a) \supset Q(g(b), c)$, $\forall x\ P(x)$, $\forall x \exists y\ P(x) \supset Q(y, x)$, ...
- Special formulae: $\perp$, $\top$

Logical language

- Ground term, atom, literal, formula: no occurrences of variables
- Closed formula: all variables are quantified (aka: sentence)

Defined symbols and free symbols

- A symbol is defined if it comes with axioms, e.g., $\simeq$
- Equality ($\simeq$) comes with the congruence axioms
- It is free otherwise, e.g., P
- Aka: interpreted/uninterpreted

Equality and the congruence axioms

- $\forall x.\ x \simeq x$
- $\forall x \forall y.\ x \simeq y \supset y \simeq x$
- $\forall x \forall y \forall z.\ x \simeq y \wedge y \simeq z \supset x \simeq z$
- $\forall x \forall y.\ x \simeq y \supset f(\ldots, x, \ldots) \simeq f(\ldots, y, \ldots)$
- $\forall x \forall y.\ [x \simeq y \wedge P(\ldots, x, \ldots)] \supset P(\ldots, y, \ldots)$

Craig interpolation or interpolation tout court

- Formulæ A and B such that $A \vdash B$
- An interpolant I is a formula that lies between A and B:
  - Derivability: $A \vdash I$ and $I \vdash B$
  - Signature: I made of symbols common to A and B
    where symbol means predicate, function, constant symbol

Trivial cases

- All symbols of A appear in B: then A itself is the interpolant
- All symbols of B appear in A: then B itself is the interpolant

Assume that at least one has at least one symbol that does not appear in the other

Craig's Interpolation Theorem (1957)

- If A and B are closed formulæ with at least one predicate symbol in common
- Then an interpolant I exists and it is also a closed formula
- No predicate symbol in common: either A is unsatisfiable and I is $\perp$ or B is valid and I is $\top$

Theorem proving

- $A \vdash^? B$ is a theorem-proving problem
- Refutational theorem proving
- Equivalently: is $A \wedge \neg B$ inconsistent?
- $A \wedge \neg B \vdash^? \bot$
- $A, \neg B \vdash^? \bot$

Proofs by refutation: reverse interpolant

- A and B inconsistent: $A, B \vdash \perp$
- Then $A \vdash I$ and $B, I \vdash \perp$
- All symbols in I common to A and B

Reverse interpolant of (A, B): interpolant of $(A, \neg B)$
because $A, B \vdash \perp$ means $A \vdash \neg B$ and $B, I \vdash \perp$ means $I \vdash \neg B$

Interpolant of (A, B): reverse interpolant of $(A, \neg B)$

In refutational settings we say interpolant for reverse interpolant

Example

- A is $\forall x.\ P(c, x)$
- B is $\forall x.\ \neg P(x, d)$
- A and B are inconsistent
- Interpolant I is $\exists y \forall x.\ P(y, x)$

Reasoning modulo theory $\mathcal{T}$

- $\vdash_{\mathcal{T}}$ in place of $\vdash$
- All uninterpreted symbols in I common to A and B
- No restrictions on interpreted symbols

Example

- A is $a_1 \not\simeq a_2$
- B is $\forall x \forall y.\ x \simeq y$
- A and B are inconsistent
- Interpolant I is $\exists x \exists y.\ x \not\simeq y$

Clausal theorem proving

- Clause: disjunction of literals where all variables are implicitly universally quantified
- $\neg P(f(z)) \vee \neg Q(g(z)) \vee R(f(z), g(z))$
- No loss of generality: every formula can be transformed into a conjunction, or set, of clauses
- Inconsistency is preserved

Transformation into clausal form

- Eliminate $\equiv$ and $\supset$: ($F \equiv G$ becomes $(F \supset G) \wedge (G \supset F)$ and $F \supset G$ becomes $\neg F \vee G$)
- Reduce the scope of all occurrences of $\neg$ to an atom: ($\neg(F \vee G)$ becomes $\neg F \wedge \neg G$, $\neg(F \wedge G)$ becomes $\neg F \vee \neg G$, $\neg\neg F$ becomes F, $\neg\exists F$ becomes $\forall\neg F$, and $\neg\forall F$ becomes $\exists\neg F$)
- Standardize variables apart
(each quantifier occurrence binds a distinct variable symbol)
- Skolemize $\exists$ and then drop $\forall$
- Distributivity and associativity: $F \vee (G \wedge H)$ becomes $(F \vee G) \wedge (F \vee H)$ and $F \vee (G \vee H)$ becomes $F \vee G \vee H$
- Replace $\wedge$ by comma and get a set of clauses

Skolemization

- Outermost $\exists$:
  - $\exists x\ F[x]$ becomes $F[a]$ (all occurrences of x replaced by a)
    a is a new Skolem constant
  - There exists an element such that F: let this element be named a
- $\exists$ in the scope of $\forall$:
  - $\forall y \exists x\ F[x, y]$ becomes $\forall y\ F[g(y), y]$
    (all occurrences of x replaced by $g(y)$)
    g is a new Skolem function
  - For all y there is an x such that F: x depends on y;
    let g be the map of this dependence

A simple example

- $\neg\{[\forall x\ P(x)] \supset [\exists y\ \forall z\ Q(y,z)]\}$
- $\neg\{\neg[\forall x\ P(x)] \vee [\exists y\ \forall z\ Q(y,z)]\}$
- $[\forall x\ P(x)] \wedge \neg[\exists y\ \forall z\ Q(y,z)]$
- $[\forall x\ P(x)] \wedge [\forall y\ \exists z\ \neg Q(y,z)]$
- $[\forall x\ P(x)] \wedge [\forall y\ \neg Q(y,f(y))]$ where f is a Skolem function
- $\{P(x),\ \neg Q(y,f(y))\}$: a set of two unit clauses

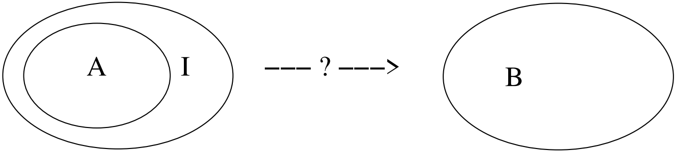
From now on we work with clauses

Why interpolation?

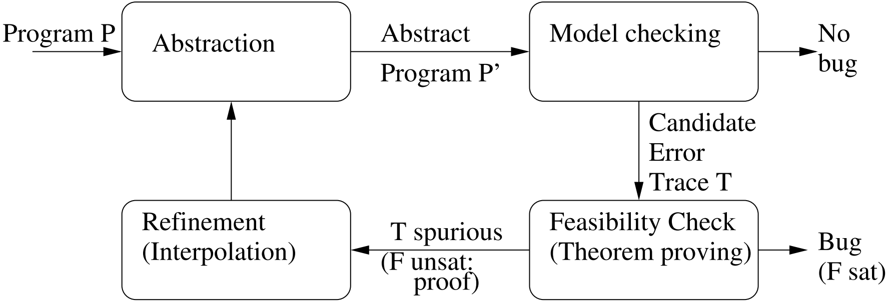
- Interpolant is a formula in between formulæ
- Formulæ represent states that satisfy them
- States of an automaton, of a transition system, of a program
- Interpolant may give information on intermediate states

Image computation in model checking

- Transition system with transition relation
- Forward reachability: computing images
- Backward reachability: computing pre-images
- Interpolant: over-approximation of an image/pre-image
- Interpolation to accelerate convergence towards fixed point



Abstraction refinement in software model checking



$F = A \cup B$; add predicates from interpolant I of (A, B): exclude T

Automated invariant generation

- Loop: *pre* `while C do T` *post*
  - $\forall s.\ pre[s] \supset I(s)$
  - $\forall s, s'.\ I(s) \wedge C[s] \wedge T[s, s'] \supset I(s')$
  - $\forall s.\ I(s) \wedge \neg C[s] \supset post(s)$
- Invariant I made of symbols common to *pre* and *post*; no symbols local to the loop body `T`
- A: k-unfolding of loop; B: post-condition violated
- $A, B \vdash \bot$
- Interpolant of (A, B): candidate invariant

Why interpolation?

- Interpolant is an explanation of $A, B \vdash \bot$
- Conflict-driven reasoning: explaining conflicts,
  where a conflict is an inconsistency between a formula to be satisfied and a candidate model

Example of explanation by interpolation I

$F = \{x \geq 2,\ \neg(x \geq 1) \vee y \geq 1,\ x^2 + y^2 \leq 1 \vee xy > 1\}$

- Caveat: x and y here are constant symbols logically
- $M = \emptyset$
- $M \; = \; x \geq 2$
- $M \; = \; x \geq 2, \;\; x \geq 1$
- $M \; = \; x \geq 2, \;\; x \geq 1, \;\; y \geq 1$
- $M \; = \; x \geq 2, \;\; x \geq 1, \;\; y \geq 1, \;\; x^2 + y^2 \leq 1$
- $M \; = \; x \geq 2, \;\; x \geq 1, \;\; y \geq 1, \;\; x^2 + y^2 \leq 1, \;\; x \leftarrow 2$
- Conflict: no value for y such that $4 + y^2 \leq 1$

Example of explanation by interpolation II

$F = \{x \geq 2,\ \neg(x \geq 1) \vee y \geq 1,\ x^2 + y^2 \leq 1 \vee xy > 1\}$

- $x^2 + y^2 \leq 1$ implies $-1 \leq x \wedge x \leq 1$ which is inconsistent with $x = 2$
- $-1 \leq x \wedge x \leq 1$ is an interpolant because x is shared
- Learn $\neg(x^2 + y^2 \leq 1) \vee x \leq 1$
- Undo $x \leftarrow 2$ and add $x \leq 1$
- $M \ = \ x \geq 2, \quad x \geq 1, \quad y \geq 1, \quad x^2 + y^2 \leq 1, \quad x \leq 1$

Interpolation in propositional logic

Interpolation in propositional logic

Terminology for interpolation: Colors

Uninterpreted symbol:

- A-colored: occurs in A and not in B
- B-colored: occurs in B and not in A
- Transparent: occurs in both

Alternative terminology: A-local, B-local, global

Terminology for interpolation: Colors

Ground term/literal/clause:

- All transparent symbols: transparent
- A-colored (at least one) and transparent symbols: A-colored
- B-colored (at least one) and transparent symbols: B-colored
- Otherwise: AB-mixed

Interpolation system

- A and B sets of clauses
- Given: a refutation of $A \cup B$
- Interpolation system: extracts interpolant of (A, B)
- How? Computing a partial interpolant $PI(C)$ for each clause C in refutation
- Defined in such a way that $PI(\Box)$ is interpolant of (A, B)

Partial interpolant

- Clause C in refutation of $A \cup B$
- $A \wedge B \vdash C$
- $A \wedge B \vdash C \vee C$
- $A \wedge \neg C \vdash \neg B \vee C$
- Interpolant of $A \wedge \neg C$ and $\neg B \vee C$
- Reverse interpolant of $A \wedge \neg C$ and $B \wedge \neg C$
- The signatures of $A \wedge \neg C$ and $B \wedge \neg C$ are not necessarily those of A and B unless C is transparent
- Use projections

Symmetric projections

C: disjunction (conjunction) of literals

- $C|_A$: A-colored and transparent literals
- $C|_B$: B-colored and transparent literals
- $C|_{A,B}$: transparent literals
- $\bot$ ($\top$) if empty

If C has no AB-mixed literals: $C = C|_A \vee C|_B$

Asymmetric projections

C: disjunction (conjunction) of literals

- $C\backslash_B = C|_A \setminus C|_{A,B}$ (A-colored only)
- $C\downarrow_B = C|_B$ (transparent go with B-colored)

If C has no AB-mixed literals: $C = C\backslash_B \vee C\downarrow_B$

Partial interpolant

- Clause C in refutation of $A \cup B$
- Partial interpolant $PI(C)$: interpolant of
  $A \wedge \neg(C|_A)$ and
  $B \wedge \neg(C|_B)$
- If C is $\Box$: $PI(C)$ interpolant of (A, B)
- Requirements:
  - $A \wedge \neg(C|_A) \vdash PI(C)$
  - $B \wedge \neg(C|_B) \wedge PI(C) \vdash \bot$
  - $PI(C)$ transparent
- Or as above with asymmetric projections

Complete interpolation system

An interpolation system is complete for an inference system if

- For all sets of clauses A and B such that $A \cup B$ is unsatisfiable
- For all refutations of $A \cup B$ by the inference system

It generates an interpolant of (A, B)

There may be more than one

Inductive approach to interpolation

- The interpolation system is defined inductively
- By defining the partial interpolant of the consequence given the partial interpolants of the premises for each inference rule
- Prove complete:
  show that its partial interpolants are indeed such

Propositional resolution: example

$$\frac{P \vee \neg Q \vee \neg R, \;\; \neg P \vee O}{O \vee \neg Q \vee \neg R}$$

where O, P, Q, and R are propositional atoms
(aka propositional variables, aka 0-ary predicates)

Propositional resolution

$$\frac{S \cup \{L \vee C,\ \neg L \vee D\}}{S \cup \{L \vee C,\ \neg L \vee D,\ C \vee D\}}$$

- L is an atom
- C and D are disjunctions of literals
- L and $\neg L$ are the literals resolved upon
- $C \vee D$ is called resolvent

First-order ground resolution

$$\frac{P(c, g(a)) \vee \neg R(c, b),\ \ \neg P(c, g(a)) \vee Q(a, g(a))}{\neg R(c, b) \vee Q(a, g(a))}$$

Same as propositional resolution: map ground atoms into propositional atoms

Example in propositional logic

$A = \{a \vee e,\ \neg a \vee b,\ \neg a \vee c\} \quad B = \{\neg b \vee \neg c \vee d,\ \neg d,\ \neg e\}$

1. $a \vee e$ resolves with $\neg e$ to yield a
2. a resolves with $\neg a \vee c$ to yield c
3. a resolves with $\neg a \vee b$ to yield b
4. b resolves with $\neg b \vee \neg c \vee d$ to yield $\neg c \vee d$
5. c resolves with $\neg c \vee d$ to yield d
6. d resolves with $\neg d$ to yield $\Box$

Goal: interpolate this refutation to get an interpolant of (A, B)

Propositional interpolation systems

- Literals in proof are input literals
- Input literals are either A-colored or B-colored or transparent
- No AB-mixed literals

The HKPYM interpolation system

C clause in refutation of $A \cup B$ by propositional resolution:

- $C \in A$: $PI(C) = \bot$
- $C \in B$: $PI(C) = \top$
- $C \vee D$ propositional resolvent of $p_1 : C \vee L$ and $p_2 : D \vee \neg L$:
  - L A-colored: $PI(C \vee D) = PI(p_1) \vee PI(p_2)$
  - L B-colored: $PI(C \vee D) = PI(p_1) \wedge PI(p_2)$
  - L transparent: $PI(C \vee D) = (L \vee PI(p_1)) \wedge (\neg L \vee PI(p_2))$

Symmetric projections

[Huang 1995] [Krajíček 1997] [Pudlàk 1997] [Yorsh, Musuvathi 2005]

Example with HKPYM

$A = \{a \vee e,\ \neg a \vee b,\ \neg a \vee c\} \quad B = \{\neg b \vee \neg c \vee d,\ \neg d,\ \neg e\}$

1. $a \vee e\ [\bot]$ resolves with $\neg e\ [\top]$ to yield $a\ [e]$:
   $PI(a) = (e \vee \bot) \wedge (\neg e \vee \top) = e$
2. $a\ [e]$ resolves with $\neg a \vee c\ [\bot]$ to yield $c\ [e]$: $PI(c) = e \vee \bot = e$
3. $a\ [e]$ resolves with $\neg a \vee b\ [\bot]$ to yield $b\ [e]$: $PI(b) = e \vee \bot = e$
4. $b\ [e]$ resolves with $\neg b \vee \neg c \vee d\ [\top]$ to yield $\neg c \vee d\ [b \vee e]$:
   $PI(\neg c \vee d) = (b \vee e) \wedge (\neg b \vee \top) = b \vee e$
5. $c\ [e]$ resolves with $\neg c \vee d\ [b \vee e]$ to yield $d\ [e \vee (c \wedge b)]$:
   $PI(d) = (c \vee e) \wedge (\neg c \vee b \vee e) = e \vee (c \wedge b)$
6. $d\ [e \vee (c \wedge b)]$ resolves with $\neg d\ [\top]$ to yield $\Box\ [e \vee (c \wedge b)]$:
   $PI(\Box) = (e \vee (c \wedge b)) \wedge \top = e \vee (c \wedge b)$

The MM interpolation system

C clause in refutation of $A \cup B$ by propositional resolution:

- $C \in A$: $PI(C) = C|_{A,B}$
- $C \in B$: $PI(C) = \top$
- $C \vee D$ propositional resolvent of $p_1: C \vee L$ and $p_2: D \vee \neg L$:
  - L A-colored: $PI(C \vee D) = PI(p_1) \vee PI(p_2)$
  - L B-colored or transparent: $PI(C \vee D) = PI(p_1) \wedge PI(p_2)$

Asymmetric projections

[McMillan 2003]

Example with MM

$A = \{a \vee e,\ \neg a \vee b,\ \neg a \vee c\}$ $\quad B = \{\neg b \vee \neg c \vee d,\ \neg d,\ \neg e\}$

1. $a \vee e\ [e]$ resolves with $\neg e\ [\top]$ to yield $a\ [e]$: $PI(a) = e \wedge \top = e$
2. $a\ [e]$ resolves with $\neg a \vee c\ [c]$ to yield $c\ [e \vee c]$: $PI(c) = e \vee c$
3. $a\ [e]$ resolves with $\neg a \vee b\ [b]$ to yield $b\ [e \vee b]$: $PI(b) = e \vee b$
4. $b\ [e \vee b]$ resolves with $\neg b \vee \neg c \vee d\ [\top]$ to yield $\neg c \vee d\ [e \vee b]$: $PI(\neg c \vee d) = (e \vee b) \wedge \top = e \vee b$
5. $c\ [e \vee c]$ resolves with $\neg c \vee d\ [e \vee b]$ to yield $d\ [e \vee (c \wedge b)]$: $PI(d) = (e \vee c) \wedge (e \vee b) = e \vee (c \wedge b)$
6. $d\ [e \vee (c \wedge b)]$ resolves with $\neg d\ [\top]$ to yield $\Box\ [e \vee (c \wedge b)]$: $PI(\Box) = (e \vee (c \wedge b)) \wedge \top = e \vee (c \wedge b)$

Comparison of HKPYM and MM

- In this example the final interpolant is the same, although at each step the HKPYM partial interpolant implies the MM partial interpolant
- In general: MM interpolants imply HKPYM interpolants [D'Silva, Kroening, Purandare, Weissenbacher 2010]
- But there is no general result as to whether weaker or stronger is preferable

Interpolation and equality

Interpolation and equality

Equational reasoning

Replacing equals by equals as in ground rewriting:

$$\frac{S \cup \{f(a,a) \simeq a,\ P(f(a,a)) \vee Q(a)\}}{S \cup \{f(a,a) \simeq a,\ P(a) \vee Q(a)\}}$$

It can be done as $f(a,a) \succ a$: replacing equals by equals needs an ordering in order to know in which direction apply the equality

Monotonicity

- $\succ$ ordering
- $s \succ t$
- Example: $f(a, i(a)) \succ e$
- Monotonicity: $r[s] \succ r[t]$ for all contexts r
  (A context is an expression, here a term or atom, with a hole)
- $f(f(a, i(a)), b) \succ f(e, b)$

Subterm property

- $\succ$ ordering
- $s[t] \succ t$
- Example: $f(a, i(a)) \succ i(a)$

Well-foundedness

- No infinite descending chain $s_0 \succ s_1 \succ \ldots s_i \succ s_{i+1} \succ \ldots$
- Monotonicity and the subterm property suffice to ensure well-foundedness on ground terms

Equality changes the picture for interpolation

- Propositional logic: no AB-mixed literals and colors are stable
- Equality: what if AB-mixed equality $t_a \simeq t_b$ is derived?
  t_a: A-colored ground term; t_b: B-colored ground term
- Rewriting: t_a and t_b in normal form, $t_a \succ t_b$:
  rewrite t_a as t_b; t_b should become transparent
- A-colored/B-colored/transparent cannot change dynamically!

Equality-interpolating theory

- (A, B): there exist transparent ground terms
- If $A \wedge B \models_{\mathcal{T}} t_a \simeq t_b$
  t_a: A-colored ground term and t_b: B-colored ground term
- Then $A \wedge B \models_{\mathcal{T}} t_a \simeq t \wedge t_b \simeq t$ for some transparent ground term t called equality-interpolating term

[Yorsh, Musuvathi 2005]

Separating ordering

Ordering $\succ$ on terms and literals:
separating if $s \succ r$ whenever r is transparent and s is not
([McMillan 2008], [Kovàcs, Voronkov 2009])

Rewriting: t_a and t_b rewritten to t

Separating implies no AB-mixed literals

- Γ: inference system with resolution, superposition, simplification, subsumption ...
- Lemma: If the ordering $\succ$ is separating, ground Γ-refutations contain no AB-mixed literals
  - $s \simeq r$ and $l[s]$ not AB-mixed, and $s \succ r$
  - either s and r same color or r transparent
  - $l[r]$ not AB-mixed

EUF is equality-interpolating

- Theorem: The quantifier-free fragment of the theory of equality is equality-interpolating
  - Γ with $\succ$ separating ordering
  - (A, B): there exist transparent ground terms
  - If $A \wedge B \models t_a \simeq t_b$
  - $A \cup B \cup \{t_a \not\simeq t_b\} \vdash_\Gamma \bot$ by refutational completeness of Γ
  - No AB-mixed equalities as $\succ$ is separating
  - Valley proof $t_a \xrightarrow{*} t \xleftarrow{*} t_b$ contains at least a transparent term
  - t must be transparent

Interpolation system GΓI

C clause in ground Γ-refutation of $A \cup B$:

- Base cases and resolution: same as in HKPYM
- $c\colon C \vee l[r] \vee D$ generated from $p_1\colon C \vee s \simeq r$ and $p_2\colon l[s] \vee D$
  - $s \simeq r$ A-colored: $PI(c) = PI(p_1) \vee PI(p_2)$
  - $s \simeq r$ B-colored: $PI(c) = PI(p_1) \wedge PI(p_2)$
  - $s \simeq r$ transparent: $PI(c) = (s \simeq r \vee PI(p_1)) \wedge (s \not\simeq r \vee PI(p_2))$

Example

$A = \{P(c),\ \neg P(e)\} \qquad B = \{c \simeq e\} \qquad c \succ e$

P is A-colored, c and e are transparent

1. $c \simeq e$ $[\top]$ simplifies $P(c)$ $[\bot]$ into $P(e)$ $[c \not\simeq e]$
   $PI(P(e)) = (c \simeq e \vee \top) \wedge (c \not\simeq e \vee \bot) = c \not\simeq e$
2. $\neg P(e)$ $[\bot]$ resolves with $P(e)$ $[c \not\simeq e]$ to yield $\Box$ $[c \not\simeq e]$
   $PI(\Box) = \bot \vee c \not\simeq e = c \not\simeq e$

Example

$$A = \{Q(f(a)),\ f(a) \simeq c\} \qquad B = \{\neg Q(f(b)),\ f(b) \simeq c\}$$

a is A-colored, b is B-colored, all other symbols are transparent

1. $f(a) \simeq c$ $[\bot]$ simplifies $Q(f(a))$ $[\bot]$ into $Q(c)$ $[\bot]$
   where $f(a) \succ c$ in any separating ordering
   $PI(Q(c)) = \bot \vee \bot = \bot$
2. $f(b) \simeq c$ $[\top]$ simplifies $\neg Q(f(b))$ $[\top]$ into $\neg Q(c)$ $[\top]$
   where $f(b) \succ c$ in any separating ordering
   $PI(\neg Q(c)) = \top \wedge \top = \top$
3. $Q(c)$ $[\bot]$ resolves with $\neg Q(c)$ $[\top]$ to yield $\Box$ $[Q(c)]$
   $PI(\Box) = (Q(c) \vee \bot) \wedge (\neg Q(c) \vee \top) = Q(c)$

Completeness

- Theorem: If the ordering is separating, $G\Gamma I$ is a complete interpolation system for ground Γ-refutations
- The proof shows that the partial interpolants built by $G\Gamma I$ satisfy the requirements for partial interpolants.

References

- Maria Paola Bonacina and Moa Johansson. Interpolation systems for ground proofs in automated deduction: a survey. Journal of Automated Reasoning, 54(4):353-390, 2015 [providing 89 references]
- Maria Paola Bonacina and Moa Johansson. Towards interpolation in an SMT solver with integrated superposition. 9th SMT Workshop, Snowbird, Utah, USA, July 2011; TR UCB/EECS-2011-80, 9-18, 2011
- Maria Paola Bonacina and Moa Johansson. On interpolation in decision procedures. In Proc. of the 20th TABLEAUX Conference, Bern, Switzerland, July 2011; Springer, LNAI 6793, 1–16, 2011

Discussion

- Generality: interpolants for more logics, theories, inference systems
- Quality: better interpolants; stronger? weaker? shorter?
- Non-ground proofs theories?
  Two-stage approach:
  Maria Paola Bonacina and Moa Johansson. On interpolation in automated theorem proving. Journal of Automated Reasoning, 54(1):69-97, 2015