Interpolation systems for ground proofs

Maria Paola Bonacina

Dipartimento di Informatica
Università degli Studi di Verona
Verona, Italy

Formal Topics Series
Computer Science Laboratory, SRI International
Menlo Park, California, USA

10 August 2016

What is interpolation?

- 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)

Closed formula: all variables are quantified (aka: sentence)

- A and B closed formulæ with at least one predicate symbol in common
- 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 $\top$

Clausal theorem proving: A and B are sets of clauses

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 $I \vdash \neg B$

In refutational settings we say interpolant for reverse interpolant

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 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 $\Box$

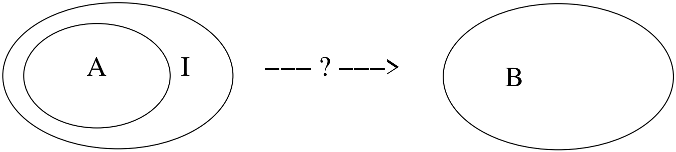
Interpolant I: $(e \vee b) \wedge (e \vee c) \equiv e \vee (b \wedge c)$

Why interpolation?

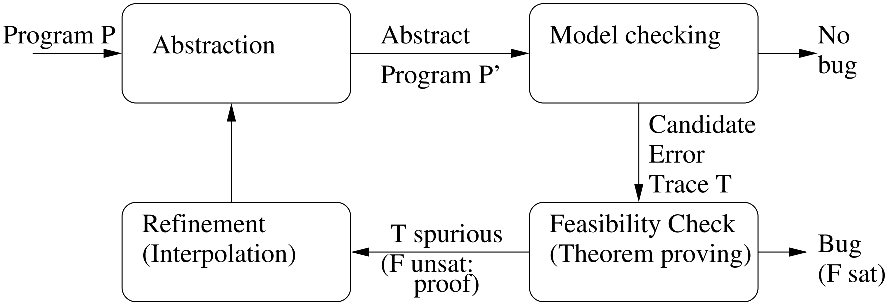
- Interpolant is a formula in between formulæ
- Formulæ represent states that satisfy them
- States of an automaton, of a transition system, of a program
- Interpolant may give information on intermediate states

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

Several approaches to interpolation

- Building interpolation into satisfiability procedures (e.g., congruence closure) [Fuchs, Goel, Grundy, Krstić, Tinelli 2012]
- Locality based [Sofronie-Stokkermans 2008]
- Via Horn clause reasoning [Gupta, Popeea, Rybalchenko 2011], [Rümmer, Hojjat, Kuncak 2013]
- Meta-rules based approach [Bruttomesso, Ghilardi, Ranise 2012], [Bruttomesso, Ghilardi, Ranise 2014]
- Inductive approach: by structural induction on the refutation

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:
  - $A \wedge \neg(C|_A) \vdash PI(C)$
  - $B \wedge \neg(C|_B) \wedge PI(C) \vdash \perp$
  - $PI(C)$ 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 all generative inference rules (e.g., superposition, simplification, not subsumption)
- Prove complete:
  show that its partial interpolants are indeed such

Interpolation for propositional resolution

- DPLL-CDCL
- Inference system Γ with resolution, superposition, simplification, subsumption ...

- If given a problem in propositional logic
- Both generate proof by resolution

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

Equality changes the picture ...

- 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
- Congruence closure: t_a and t_b representatives of singly-colored classes: merge: one of them should become transparent
- 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

- $\mathcal{T}$: convex 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

Congruence closure: t representative of the new congruence class

[Yorsh, Musuvathi 2005]

Separating ordering

Ordering $\succ$ on terms and literals:
separating if $s \succ r$ whenever r is transparent and s is not

Rewriting: t_a and t_b rewritten to t

[McMillan 2008], [Kovàcs, Voronkov 2009]

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 $t_a \stackrel{*}{\rightarrow} t \stackrel{*}{\leftarrow} t_b$ contains at least a transparent term
  - t must be transparent

Other convex equality-interpolating theories

- Non-empty lists
- Linear rational arithmetic:
  - $A \wedge B \supset a \simeq b$
  - $A \wedge B \supset a \leq b \wedge b \leq a$
  - $\exists\ t_1$ such that $A \wedge B \supset a \leq t_1 \leq b$
  - $\exists\ t_2$ such that $A \wedge B \supset b \leq t_2 \leq a$
  - $A \wedge B \supset a \simeq t_1 \simeq t_2 \simeq b$

[Yorsh, Musuvathi 2005]

Equality sharing aka Nelson-Oppen method

$\mathcal{T}$-satisfiability procedure for $\mathcal{T} = \bigcup_{i=1}^{n} \mathcal{T}_i$

- Disjoint, convex, equality-interpolating theories
- Equipped with $\mathcal{T}_i$-satisfiability procedure $\mathcal{Q}_i$ that generate equality-interpolating terms, proofs, and $\mathcal{T}_i$-interpolants
- S input set of ground $\mathcal{T}$-literals
- Partition $S = A \cup B$ and separation $S_1, \ldots, S_n$ are orthogonal: new free constants inherit the color of the term they replace, since there are no AB-mixed input terms

Interpolation in equality sharing

- Each $\mathcal{Q}_i$ takes as input $S_i = A_i \cup B_i$ and deals with $A_i \cup B_i \cup K$ where K contains the propagated equalities
- Equality-interpolating: K contains no AB-mixed equalities
- The proof by equality sharing contains no AB-mixed literals
- What is the partial interpolant for a propagated equality?
- Theory-specific partial interpolant

Theory-specific partial interpolant

- Propagated literal: $A_i \cup B_i \cup K \vdash_{\mathcal{T}_i} L$
  where L is either an equality or $\Box$
- Interpolation wrt partition (A', B') of $A_i \cup B_i \cup K$
  $A' = A_i \cup K \backslash_B$
  $B' = B_i \cup K \downarrow_B$
- $PI^i_{(A',B')}(L)$ is the $\mathcal{T}_i$-interpolant of
  $(A' \wedge \neg(L\backslash_B), B' \wedge \neg(L \downarrow_B))$

[Yorsh, Musuvathi 2005]

The YM interpolation system

C unit clause in refutation of $A \cup B$ by equality sharing

- $C \in A$: $PI(C) = \bot$ $\quad$ $C \in B$: $PI(C) = \top$
- C derived as $A_i \cup B_i \cup K \vdash_{\mathcal{T}_i} C$:
  $PI(C) = (PI^i_{(A',B')}(C) \vee \bigvee_{L \in A'} PI(L)) \wedge \bigwedge_{L \in B'} PI(L)$

If $K = \emptyset$ (only one theory or C does not depend on propagated equalities): $PI(C) = PI^i_{(A',B')}(C)$

Example in theory combination

$$A = \{f(x_1) + x_2 \simeq x_3, \quad f(y_1) + y_2 \simeq y_3, \quad y_1 \leq x_1\}$$

$$B = \{x_2 \simeq g(b), \quad y_2 \simeq g(b), \ x_1 \leq y_1, \quad x_3 < y_3\}$$

Let EUF be $\mathcal{T}_1$ with procedure $\mathcal{Q}_1$ and
LRA be $\mathcal{T}_2$ with procedure $\mathcal{Q}_2$

[Yorsh, Musuvathi 2005]

Example after separation

$A_1 = \{a_1 \simeq f(x_1),\quad a_2 \simeq f(y_1)\}$

$A_2 = \{a_1 + x_2 \simeq x_3,\quad a_2 + y_2 \simeq y_3,\quad y_1 \leq x_1\}$

$B_1 = \{x_2 \simeq g(b),\quad y_2 \simeq g(b)\}$

$B_2 = \{x_1 \leq y_1,\quad x_3 < y_3\}$

Shared constants: $V = \{a_1, x_1, a_2, y_1, x_2, y_2\}$

$\{f, a_1, a_2\}$ are A-colored

$\{g, b\}$ are B-colored

$\{x_1, y_1, x_2, y_2, x_3, y_3\}$ are transparent

Example: first proof step

- Q_2 deduces $x_1 \simeq y_1$ from $y_1 \leq x_1$ $[\perp]$ and $x_1 \leq y_1$ $[\top]$
- $x_1, y_1 \in V$: $x_1 \simeq y_1$ is propagated
- $A' = A_2$ and $B' = B_2$ since $K = \emptyset$
- $A' \wedge \neg((x_1 \simeq y_1)\backslash_B) = A_2 \wedge \top = A_2$
  $B' \wedge \neg((x_1 \simeq y_1)\downarrow_B) = B_2 \cup \{x_1 \not\simeq y_1\}$
- $PI^2_{(A',B')}(x_1 \simeq y_1) = y_1 \leq x_1$
  which follows from $y_1 \leq x_1 \in A_2$ and is $\mathcal{T}_2$-inconsistent with $\{x_1 \leq y_1,\ x_1 \not\simeq y_1\}$ where $x_1 \leq y_1 \in B_2$
- $PI(x_1 \simeq y_1) = y_1 \leq x_1$

Example: second proof step

- Q_1 deduces $a_1 \simeq a_2$ from $a_1 \simeq f(x_1)$ $[\perp]$, $a_2 \simeq f(y_1)$ $[\perp]$ and $x_1 \simeq y_1$ $[y_1 \leq x_1]$
- $a_1, a_2 \in V$: $a_1 \simeq a_2$ is propagated
- $A' = A_1$ and $B' = B_1 \cup \{x_1 \simeq y_1\}$ since $K = \{x_1 \simeq y_1\}$
- $A' \wedge \neg((a_1 \simeq a_2)\backslash_B) = A_1 \cup \{a_1 \not\simeq a_2\}$
  $B' \wedge \neg((a_1 \simeq a_2) \downarrow_B) = B_1 \cup \{x_1 \simeq y_1\}$
- $PI^1_{(A',B')}(a_1 \simeq a_2) = x_1 \not\simeq y_1$
  which follows from $\{a_1 \simeq f(x_1),\ a_2 \simeq f(y_1),\ a_1 \not\simeq a_2\}$ and is inconsistent with $\{x_1 \simeq y_1\}$
- $PI(a_1 \simeq a_2) = (x_1 \not\simeq y_1 \vee \perp) \wedge y_1 \leq x_1 = y_1 < x_1$

Example: third proof step

- Q_1 deduces $x_2 \simeq y_2$ from $x_2 \simeq g(b)$ $[\top]$ and $y_2 \simeq g(b)$ $[\top]$
- $x_2, y_2 \in V$: $x_2 \simeq y_2$ is propagated
- $A' = A_1$ and $B' = B_1$ since $K = \emptyset$
- $A' \wedge \neg((x_2 \simeq y_2)\backslash_B) = A_1 \wedge \top = A_1$
  $B' \wedge \neg((x_2 \simeq y_2) \downarrow_B) = B_1 \cup \{x_2 \not\simeq y_2\}$
- $PI^1_{(A',B')}(x_2 \simeq y_2) = \top$
  because $B_1 \cup \{x_2 \not\simeq y_2\}$ is $\mathcal{T}_1$-inconsistent
- $PI(x_2 \simeq y_2) = \top$

Example: fourth proof step

- Q_2 deduces $\Box$ from $a_1 + x_2 \simeq x_3\ [\bot]$, $a_2 + y_2 \simeq y_3\ [\bot]$, $x_3 < y_3\ [\top]$, $a_1 \simeq a_2\ [y_1 < x_1]$ and $x_2 \simeq y_2\ [\top]$
- $A' = A_2 \cup \{a_1 \simeq a_2\}$ and $B' = B_2 \cup \{x_2 \simeq y_2\}$ as $K = \{a_1 \simeq a_2, x_2 \simeq y_2\}$
- $A' \wedge \neg((\Box)\backslash_B) = A_2 \cup \{a_1 \simeq a_2\} \wedge \top = A_2 \cup \{a_1 \simeq a_2\}$
  $B' \wedge \neg((\Box)\downarrow_B) = B_2 \cup \{x_2 \simeq y_2\} \wedge \top = B_2 \cup \{x_2 \simeq y_2\}$
- $PI^2_{(A',B')}(\Box) = x_3 - x_2 \simeq y_3 - y_2$
  because $\{a_1 + x_2 \simeq x_3,\ a_2 + y_2 \simeq y_3,\ a_1 \simeq a_2\}$ entail $x_3 - x_2 \simeq y_3 - y_2$ which is $\mathcal{T}_2$-inconsistent with $\{x_3 < y_3,\ x_2 \simeq y_2\}$ where $x_3 < y_3 \in B_2$
- $PI(\Box) = (x_3 - x_2 \simeq y_3 - y_2 \vee y_1 < x_1) \wedge \top = x_3 - x_2 \simeq y_3 - y_2 \vee y_1 < x_1$

Interpolation in DPLL($\mathcal{T}$)

- $A \cup B$ set of ground $\mathcal{T}$-clauses
- DPLL($\mathcal{T}$)-refutation of $A \cup B$: propositional resolution + $\mathcal{T}$-lemmas ($\mathcal{T}$-conflict clauses are $\mathcal{T}$-lemmas)
- If clause C is a $\mathcal{T}$-lemma, $\neg C$ is a $\mathcal{T}$-unsatisfiable set of ground $\mathcal{T}$-literals
- No *AB*-mixed literals: $\neg C = (\neg C) \setminus_B \wedge (\neg C) \downarrow_B$
- The $\mathcal{T}$-interpolant of $((\neg C)\setminus_B, (\neg C) \downarrow_B)$ computed by YM provides partial interpolant of C in DPLL($\mathcal{T}$)-refutation

HKPYM–T and MM–T interpolation systems

Add one case to either HKPYM or MM:

- C is a $\mathcal{T}$-lemma:
  $PI(C)$ is $\mathcal{T}$-interpolant of $((\neg C)\backslash_B, (\neg C)\downarrow_B)$ extracted by YM from $\neg C \vdash_{\mathcal{T}} \bot$

Completeness: from that of HKPYM or MM and YM

[Yorsh and Musuvathi 2005]

Why interpolation for superposition?

- Superposition-based decision procedures
- DPLL($\Gamma+\mathcal{T}$): DPLL($\mathcal{T}$) with superposition (Γ) integrated for a fully automated treatment of quantifiers

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: $PI(c) = (s \simeq r \vee PI(p_1)) \wedge (s \not\simeq r \vee PI(p_2))$
- Superposition into equational literal and Simplification: same

Example with superposition

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

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 $\Box$ $[c \not\simeq e]$
   $PI(\Box) = \bot \vee c \not\simeq e = c \not\simeq e$

Another example with superposition

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

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 $\square$ $[Q(c)]$
   $PI(\square) = (Q(c) \vee \bot) \wedge (\neg Q(c) \vee \top) = Q(c)$

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.

Summary

- Survey of interpolation systems for ground refutations:
  - Unified framework of definitions for interpolation
  - Interpolation systems for propositional resolution
  - Interpolation and equality: connecting equality-interpolating theory and separating ordering
  - Interpolation system for equality sharing
  - Interpolation systems for DPLL($\mathcal{T}$)
- A complete interpolation system for ground refutations by superposition

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, non-convex 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