Interpolation in First-Order Logic with Equality

Master Thesis Presentation

Bernhard Mallinger

Advisor: Stefan Hetzl

Institute of Discrete Mathematics and Geometry
TU Wien

13. Oktober 2014

Agenda

Agenda

Introduction

- Want concrete algorithms for FOL/EQ
  $\Rightarrow$ Little attention so far
- Present different constructive proofs

Agenda

Craig Interpolation (1/2)

Theorem ([Craig, 1957]). *Let Γ and Δ be sets of first-order formulas where*

- Γ *contains red and gray symbols and*
- Δ *contains blue and gray symbols*

such that:

- $\Gamma \models \Delta$

Then there is a interpolant I containing only gray symbols such that:

- $\Gamma \models I$
- $I \models \Delta$

$$
\begin{tikzcd}
\Gamma \arrow[rr] \arrow[dr] & & \Delta \\
& I \arrow[ur] &
\end{tikzcd}
$$

Craig Interpolation (2/2)

Example

- Let $\Gamma = \{P(a))\}$ and $\Delta = \{\forall x(P(x) \supset Q(x)), \exists y Q(y)\}$.
- Interpolant: $\exists z\, P(z)$

Example

- Let $\Gamma = \{P(a), \neg P(b)\}$ and $\Delta = \{a \neq b\}$.
- Only possible interpolant: $a \neq b$

Example

- Let $\Gamma = \{P(a), \neg P(a)\}$, $\Delta = \emptyset$.
- Only possible interpolant: $\perp$

Craig Interpolation (2/2)

Example

- Let $\Gamma = \{P(a))\}$ and $\Delta = \{\forall x(P(x) \supset Q(x)), \exists y Q(y)\}$.
- Interpolant: $\exists z\, P(z)$

Example

- Let $\Gamma = \{P(a), \neg P(b)\}$ and $\Delta = \{a \neq b\}$.
- Only possible interpolant: $a \neq b$

Example

- Let $\Gamma = \{P(a), \neg P(a)\}$, $\Delta = \emptyset$.
- Only possible interpolant: $\perp$

Craig Interpolation (2/2)

Example

- Let $\Gamma = \{P(a))\}$ and $\Delta = \{\forall x(P(x) \supset Q(x)), \exists y Q(y)\}$.
- Interpolant: $\exists z\, P(z)$

Example

- Let $\Gamma = \{P(a), \neg P(b)\}$ and $\Delta = \{a \neq b\}$.
- Only possible interpolant: $a \neq b$

Example

- Let $\Gamma = \{P(a), \neg P(a)\}$, $\Delta = \emptyset$.
- Only possible interpolant: $\perp$

Craig Interpolation (2/2)

Example

- Let $\Gamma = \{P(a))\}$ and $\Delta = \{\forall x(P(x) \supset Q(x)), \exists y Q(y)\}$.
- Interpolant: $\exists z\, P(z)$

Example

- Let $\Gamma = \{P(a), \neg P(b)\}$ and $\Delta = \{a \neq b\}$.
- Only possible interpolant: $a \neq b$

Example

- Let $\Gamma = \{P(a), \neg P(a)\}$, $\Delta = \emptyset$.
- Only possible interpolant: $\perp$

Craig Interpolation (2/2)

Example

- Let $\Gamma = \{P(a))\}$ and $\Delta = \{\forall x(P(x) \supset Q(x)), \exists y Q(y)\}$.
- Interpolant: $\exists z\, P(z)$

Example

- Let $\Gamma = \{P(a), \neg P(b)\}$ and $\Delta = \{a \neq b\}$.
- Only possible interpolant: $a \neq b$

Example

- Let $\Gamma = \{P(a), \neg P(a)\}$, $\Delta = \emptyset$.
- Only possible interpolant: $\bot$

Craig Interpolation (2/2)

Example

- Let $\Gamma = \{P(a))\}$ and $\Delta = \{\forall x(P(x) \supset Q(x)), \exists y Q(y)\}$.
- Interpolant: $\exists z\, P(z)$

Example

- Let $\Gamma = \{P(a), \neg P(b)\}$ and $\Delta = \{a \neq b\}$.
- Only possible interpolant: $a \neq b$

Example

- Let $\Gamma = \{P(a), \neg P(a)\}$, $\Delta = \emptyset$.
- Only possible interpolant: $\bot$

Applications

- Proof of Beth's Definability Theorem
- Model checking
- Detecting loop invariants
- Reasoning with large knowledge bases

Agenda

Proof by reduction

Reduction to FOL without equality and function symbols:

Translate equality and function symbols:

$$\begin{aligned}\Big(P(c)\Big)^* &\equiv \exists x(C(x) \wedge P(x))\\ \Big(P(f(c))\Big)^* &\equiv \exists x(\exists y(C(y) \wedge F(y,x)) \wedge P(x))\\ \Big(s = t\Big)^* &\equiv E(s,t)\end{aligned}$$

Add axioms for equality and new predicate symbols:

$$\varphi \quad \to \quad \Big(\mathsf{T}_E \wedge \bigwedge_{f \in \mathsf{FS}} \mathsf{T}_f\Big) \supset \varphi^*$$

Clearly φ and φ^* are equisatisfiable.

Proof in FOL without equality and FS

Lemma (Maehara)

*Let Γ and Δ be sets of first-order formulas without equality and function symbols such that $\Gamma \vdash \Delta$ is provable in **sequent calculus**. Then for any partition $\langle(\Gamma_1; \Delta_1), (\Gamma_2; \Delta_2)\rangle$ there is an interpolant I such that*

1. $\Gamma_1 \vdash \Delta_1, I$ *is provable*
2. $\Gamma_2, I \vdash \Delta_2$ *is provable*
3. $\mathrm{L}(I) \subseteq \mathrm{L}(\Gamma_1, \Delta_1) \cap \mathrm{L}(\Gamma_2, \Delta_2)$

[Baaz and Leitsch, 2011] presents a strengthening which includes function symbols.

Open question: Can it be extended to include equality?

Proof in FOL without equality and FS

Lemma (Maehara)

Let Γ and Δ be sets of first-order formulas without equality and function symbols such that $\Gamma \vdash \Delta$ is provable in ***sequent calculus****. Then for any partition $\langle(\Gamma_1; \Delta_1), (\Gamma_2; \Delta_2)\rangle$ there is an interpolant I such that*

1. $\Gamma_1 \vdash \Delta_1, I$ *is provable*
2. $\Gamma_2, I \vdash \Delta_2$ *is provable*
3. $\mathrm{L}(I) \subseteq \mathrm{L}(\Gamma_1, \Delta_1) \cap \mathrm{L}(\Gamma_2, \Delta_2)$

[Baaz and Leitsch, 2011] presents a strengthening which includes function symbols.

Open question: Can it be extended to include equality?

Agenda

Interpolant extraction

Motivation

- Proof by reduction is impractical
- Goal: Compute interpolants from proof
- The following is based on [Huang, 1995]

Interpolant extraction from resolution proofs

- Skolemisation and clausal form transformation do no alter the set of interpolants
- Have to use "reverse" (but equivalent) formulation of interpolation

Interpolant extraction

Motivation

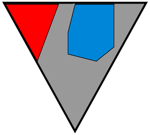
- Proof by reduction is impractical
- Goal: Compute interpolants from proof
- The following is based on [Huang, 1995]

Interpolant extraction from resolution proofs

- Skolemisation and clausal form transformation do no alter the set of interpolants
- Have to use "reverse" (but equivalent) formulation of interpolation

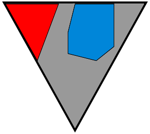
Huang's algorithm (1/2) (6 min)

Proof:



$\Downarrow$ *Extract propositional interpolant structure from proof*

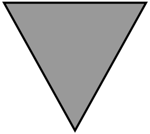
Propositional Interpolant:



$\ldots Q(f(c), c) \ldots$

$\Downarrow$ *Replace colored function and constant symbols*

Prenex First-Order Interpolant:



$\exists x_3 \forall x_5 \ldots Q(x_5, x_3) \ldots$

Huang's algorithm (2/2)

- Propositional interpolant is interpolant modulo function and constant symbols
- For the lifting phase, the ordering of the lifting variables is crucial
- The type of the quantifier is determined by the coloring of the symbol

Theorem

The number of quantifier alternations in the resulting interpolant directly corresponds to the number of color alternations of terms in the resolution proof.

Huang's algorithm (2/2)

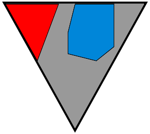
- Propositional interpolant is interpolant modulo function and constant symbols
- For the lifting phase, the ordering of the lifting variables is crucial
- The type of the quantifier is determined by the coloring of the symbol

Theorem

The number of quantifier alternations in the resulting interpolant directly corresponds to the number of color alternations of terms in the resolution proof.

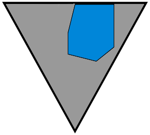
Interpolation extraction in one phase

Proof:



$\Downarrow$ *Combined structure extraction and replacing of colored symbols*

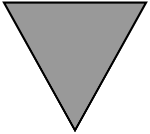
Interpolant modulo current clause:



$\forall x_5 \ldots Q(x_5, c) \ldots$

Recursively applied to all inferences of the proof results in:

Non-Prenex First-Order Interpolant:



$\exists x_3 \ldots \forall x_5 \ldots Q(x_5, x_3) \ldots$

Agenda

Semantic Proof

TODO

Agenda

Conclusion

- Craig's and Huang's proof based interpolant extraction from proofs
  $\Rightarrow$ differ in applicability
- Craig shows that the interpolation theorem holds also in FOL/EQ
- Huang shows that interpolants can efficiently be extracted in FOL/EQ
  - Does not require different methods
  - Little attention so far in research
- Interpolation also allows for a model theoretic approach

Baaz, M. and Leitsch, A. (2011).
Methods of Cut-Elimination.
Trends in Logic. Springer.

Craig, W. (1957).
Linear Reasoning. A New Form of the Herbrand-Gentzen Theorem.
Journal of Symbolic Logic, 22(3):250–268.

Huang, G. (1995).
Constructing Craig Interpolation Formulas.
In *Proceedings of the First Annual International Conference on Computing and Combinatorics*, COCOON '95, pages 181–190, London, UK, UK. Springer-Verlag.