

Interpolation in First-Order Logic with Equality

Master Thesis Presentation

Bernhard Mallinger

Advisor: Stefan Hetzl

Institute of Discrete Mathematics and Geometry
TU Wien

October 13, 2014

Agenda

- 1 Introduction
- 2 Proof by Reduction
- 3 Interpolant Extraction
- 4 Semantic Proof
- 5 Conclusion

Agenda

- 1 Introduction
- 2 Proof by Reduction
- 3 Interpolant Extraction
- 4 Semantic Proof
- 5 Conclusion

Craig Interpolation (1/2)

Theorem ([Craig, 1957]).

Let Γ and Δ be finite sets of first-order formulas where

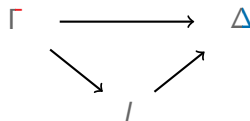
- Γ contains *red* and gray symbols and
- Δ contains *blue* and gray symbols

such that:

- $\Gamma \models \Delta$

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

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



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 $\Delta = \{a \neq b\}$.
- Interpolant: $a \neq b$

Example

- Let $\Gamma = \{P(a), \neg P(a)\}$, $\Delta = \emptyset$.
- 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 $\Delta = \{a \neq b\}$.
- Interpolant: $a \neq b$

Example

- Let $\Gamma = \{P(a), \neg P(a)\}$, $\Delta = \emptyset$.
- 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 $\Delta = \{a \neq b\}$.
- Interpolant: $a \neq b$

Example

- Let $\Gamma = \{P(a), \neg P(a)\}$, $\Delta = \emptyset$.
- 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 $\Delta = \{a \neq b\}$.
- Interpolant: $a \neq b$

Example

- Let $\Gamma = \{P(a), \neg P(a)\}$, $\Delta = \emptyset$.
- 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 $\Delta = \{a \neq b\}$.
- Interpolant: $a \neq b$

Example

- Let $\Gamma = \{P(a), \neg P(a)\}$, $\Delta = \emptyset$.
- 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 $\Delta = \{a \neq b\}$.
- Interpolant: $a \neq b$

Example

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

Applications and Motivation

Applications

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

Motivation

- Craig interpolation in full first-order logic with equality has received little attention so far
- Interest for constructive proofs giving rise to interpolant extraction algorithms

Applications and Motivation

Applications

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

Motivation

- Craig interpolation in full first-order logic with equality has received little attention so far
- Interest for constructive proofs giving rise to interpolant extraction algorithms

Agenda

- 1 Introduction
- 2 Proof by Reduction
- 3 Interpolant Extraction
- 4 Semantic Proof
- 5 Conclusion

Proof by Reduction due to Craig

Reduction to FOL without equality and function symbols:

$$\left(P(\textcolor{red}{c})\right)^* \equiv \exists x(\textcolor{red}{C}(x) \wedge P(x))$$

$$\left(P(\textcolor{blue}{f}(\textcolor{red}{c}))\right)^* \equiv \exists x(\exists y(\textcolor{red}{C}(y) \wedge \textcolor{blue}{F}(y, x)) \wedge P(x))$$

$$(s = t)^* \equiv E(s, t)$$

$$(\varphi)^* \equiv \left(\textcolor{teal}{T}_E \wedge \bigwedge_{f \in \text{FS}} \textcolor{red}{T}_{\textcolor{blue}{F}}\right) \supset \varphi^*$$

Clearly φ and φ^* are equisatisfiable.

Proof in FOL without Equality and Function Symbols

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 $\langle (\Gamma_1; \Delta_1), (\Gamma_2; \Delta_2) \rangle$ with $\Gamma_1 \uplus \Gamma_2 = \Gamma$ and $\Delta_1 \uplus \Delta_2 = \Delta$ 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 $L(I) \subseteq L(\Gamma_1, \Delta_1) \cap L(\Gamma_2, \Delta_2)$

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

Open question: Can it be extended to include equality rules for LK?

Proof in FOL without Equality and Function Symbols

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 $\langle (\Gamma_1; \Delta_1), (\Gamma_2; \Delta_2) \rangle$ with $\Gamma_1 \uplus \Gamma_2 = \Gamma$ and $\Delta_1 \uplus \Delta_2 = \Delta$ 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 $L(I) \subseteq L(\Gamma_1, \Delta_1) \cap L(\Gamma_2, \Delta_2)$

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

Open question: Can it be extended to include equality rules for LK?

Agenda

- 1 Introduction
- 2 Proof by Reduction
- 3 Interpolant Extraction**
- 4 Semantic Proof
- 5 Conclusion

Interpolant Extraction from Resolution Proofs

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 not alter the set of interpolants
- Have to use “reverse” (but equivalent) formulation of interpolation

Interpolant Extraction from Resolution Proofs

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 not alter the set of interpolants
- Have to use “reverse” (but equivalent) formulation of interpolation

The Resolution Calculus

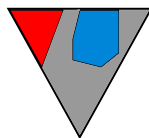
$$\textit{Resolution:} \quad \frac{C \vee I \quad D \vee \neg I'}{(C \vee D)\sigma} \text{ res} \quad \sigma = \text{mgu}(I, I')$$

$$\textit{Factorization:} \quad \frac{C \vee I \vee I'}{(C \vee I)\sigma} \text{ fac} \quad \sigma = \text{mgu}(I, I')$$

$$\textit{Paramodulation:} \quad \frac{D \vee s = t \quad E[r]_p}{(D \vee E[t]_p)\sigma} \text{ par} \quad \sigma = \text{mgu}(s, r)$$

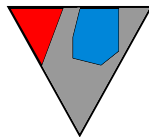
Huang's Algorithm (1/3)

Proof:



*Extract propositional interpolant
structure from proof*

Propositional
Interpolant:

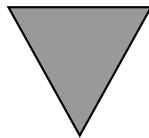


$\dots Q(\textcolor{red}{f}(\textcolor{blue}{c}), \textcolor{blue}{c}) \dots$



*Replace colored function and
constant symbols*

Prenex
First-Order
Interpolant:



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

Propositional Interpolant Extraction Example

Paramodulation rule:

$$\frac{C_1 : D \vee s = t \quad C_2 : E[r]_p}{C : (D \vee E[t]_p)\sigma} \text{ par} \quad \sigma = \text{mgu}(s, r)$$

Propositional interpolant¹:

$$\text{PI}(C) \stackrel{\text{def}}{=} [(s = t \wedge \text{PI}(C_2)) \vee (s \neq t \wedge \text{PI}(C_1))]\sigma$$

¹Provided that r is not contained in a colored term

Huang's Algorithm (2/3)

TODO: remove from here as soon as it's clear if it's being said at the image slide

First phase

- Propositional interpolant is extracted inductively, is boolean combination of PIs of clauses and resolved literals or equations of paramodulation inferences.
- Propositional interpolant is interpolant modulo function and constant symbols (only grey predicate symbols) (this strategy already gives rise to a complete procedure for propositional logic)
- Rule for paramodulation somewhat more complex but still same approach as for resolution and factorisation

Huang's Algorithm (3/3)

TODO: remove from here as soon as it's clear if it's being said at the image slide

Second phase

- The second phase replaces the remaining colored terms by quantified variables
- The ordering of the lifting variables is crucial
- The type of the quantifier is determined by the coloring of the symbol

KEEP THIS:

- Number of quantifier alternations \sim number of color alternations in terms of the resolution proof

Huang's Algorithm (3/3)

TODO: remove from here as soon as it's clear if it's being said at the image slide

Second phase

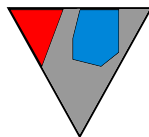
- The second phase replaces the remaining colored terms by quantified variables
- The ordering of the lifting variables is crucial
- The type of the quantifier is determined by the coloring of the symbol

KEEP THIS:

- Number of quantifier alternations \sim number of color alternations in terms of the resolution proof

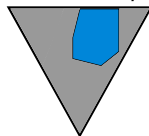
Interpolation Extraction in one Phase

Proof:



*Combined structure extraction and
replacing of colored symbols*

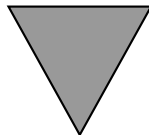
Interpolant
modulo
current clause:



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

*Recursively applied to all infer-
ences of the proof results in:*

Non-Prenex
First-Order
Interpolant:



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

Agenda

- 1 Introduction
- 2 Proof by Reduction
- 3 Interpolant Extraction
- 4 Semantic Proof**
- 5 Conclusion

Semantic Proof

- Indirect and model-theoretic proof of the interpolation theorem
- Inherently non-constructive
- Equality does not require explicit handling
- Interpolation in FOL with equality equivalent to Robinson's joint consistency theorem

Agenda

- 1 Introduction
- 2 Proof by Reduction
- 3 Interpolant Extraction
- 4 Semantic Proof
- 5 Conclusion**

Conclusion

- Craig's and Huang's proof based interpolant extraction from proofs
⇒ but differ in practical applicability
- Craig shows that the interpolation theorem holds also in FOL with equality
- Huang shows that interpolants can efficiently be extracted in FOL with equality
 - Handling of equality does not require a different approach
 - Little attention so far in research
- Huang's two-stage approach can be changed to a one-stage approach yielding non-prenex interpolants
- Interpolation also allows for a model theoretic approach

References



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.