FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

Interpolation in First-Order Logic with Equality

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Computational Intelligence

eingereicht von

Bernhard Mallinger
Matrikelnummer 0707663

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuung: Ass.Prof. Stefan Hetzl
Mitwirkung: Dr. Vorname Familienname
Dr. Vorname Familienname

Wien, TT.MM.JJJJ

(Unterschrift Verfasser) (Unterschrift Betreuung)

FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

Interpolation in First-Order Logic with Equality

MASTER'S THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Computational Intelligence

by

Bernhard Mallinger
Registration Number 0707663

to the Faculty of Informatics
at the Vienna University of Technology

Advisor: Ass.Prof. Stefan Hetzl
Assistance: Dr. Vorname Familienname
Dr. Vorname Familienname

Vienna, TT.MM.JJJJ

(Signature of Author) (Signature of Advisor)

Erklärung zur Verfassung der Arbeit

Bernhard Mallinger
Gassergasse 25/17-18, 1050 Wien

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit - einschließlich Tabellen, Karten und Abbildungen -, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

(Ort, Datum) (Unterschrift Verfasser)

Acknowledgements

Optional acknowledgements may be inserted here.

Abstract

According to the guidelines of the faculty, an abstract in English has to be inserted here.

Kurzfassung

Contents

Chapter 1

Introduction

1.1 Preliminaries

this section contains all the required notation but will just be written up nicely at the end

The language of a first-order formula A is denoted by $\mathrm{L}(A)$ and contains all predicate, constant and function symbols that occur in A. These are also referred to as the *non-logical symbols* of A. The *logical symbols* on the other hand include all logical connectives, quantifiers, the equality symbol ($=$) as well as symbols denoting truth ($\top$) and falsity ($\bot$).

For formulas $A_1, \ldots, A_n$, $\mathrm{L}(A_1, \ldots, A_n) = \bigcup_{1 \leqslant i \leqslant n} \mathrm{L}(A_i)$.

A term s is a subterm of a term t if s occurs in t. s is a strict subterm of t if s is a subterm of t and $s \neq t$.

An occurrence of Φ-term is called *maximal* if it does not occur as subterm of another Φ-term. An occurrence of a colored term t is a maximal colored term if it does not occur as subterm of another colored term.

colors are only defined later

We denote $x_1, \ldots, x_n$ by $\bar{x}$.

For a set of formulas Φ, $\neg\Phi$ denotes $\{\neg A \mid A \in \Phi\}$.

A substitution is a mapping of variables to terms. It is denoted by $\phi[x/t]$, where ϕ is a formula or term where each occurrence of the variable x is replaced by the term t. A substitution σ is called trivial on x if $x\sigma = x$. Otherwise it is called non-trivial.

An *abstraction* on the other hand is a mapping of terms to variables. It is denoted by $\phi\{t/x\}$, where ϕ is a formula or term where each occurrence of the term t is replaced by the variable x.

A term s is an *instance* of a term t if there exists a substitution σ such that $t\sigma = s$. If s is an instance of t, then t is an *abstraction* of s. Note that the abstraction- and instance-relation are reflexive. s is a *proper* instance (abstraction) of t if s is an instance (abstraction) of t and $s \neq t$.

The length of a term or formula ϕ is the number of logical and non-logical symbols in ϕ.

$A[s]_p$ denotes A with an occurrence of s at position p.

$A[s]$ denotes A where the term s occurs on some set of positions Φ. $A[t]$ denotes $A[s]$ where on each position in Φ, s has been replaced by t. Due to its vagueness, this notation is mostly used in order to emphasis that the term s does occur in A in some way.

TODO: define Σ as subformula set; possibly remove definition in chapter 2

TODO: define $\equiv$ as syntactic equality. Define also $\Leftrightarrow$, $\leftrightarrow$.

TODO: define what we mean by model and free variables. (need universal quantification of free vars)

TODO: define ground, non-ground

TODO: define set-notation of unifiers

TODO: define infinite-domain unifiers

TODO: define range and domain of substitutions

TODO: define prenex formulas with matrix and prefix

1.2 Craig Interpolation

TODO: write some text about what interpolation means and that we prove more or less only reverse interpolation, but that's fine by the proposition

Definition 1.1. Let Γ and Δ be sets of first-order formulas. An *interpolant* of Γ and Δ is a first-order formula I such that

1. $\Gamma \models I$
2. $I \models \Delta$
3. $\mathrm{L}(I) \subseteq \mathrm{L}(\Gamma) \cap \mathrm{L}(\Delta)$.

A *reverse interpolant* of Γ and Δ is a first-order formula I such that I meets conditions 1 and 3 of an interpolant as well as:

2'. $\Delta \models \neg I$ △

Theorem 1.2 (Interpolation)**.** *Let Γ and Δ be sets of first-order formulas such that $\Gamma \models \Delta$. Then there exists an interpolant for Γ and Δ.*

Theorem 1.3 (Reverse Interpolation)**.** *Let Γ and Δ be sets of first-order formulas such that $\Gamma \cup \Delta$ is unsatisfiable. Then there exists a reverse interpolant for Γ and Δ.*

Proposition 1.4. *Theorem 1.2 and 1.3 are equivalent.*

Proof. Let Γ and Δ be sets of first-order formulas such that $\Gamma \models \Delta$. Then $\Gamma \cup \neg\Delta$ is unsatisfiable. By Theorem 1.3, there exists a reverse interpolant I for Γ and $\neg\Delta$. As $\neg\Delta \models \neg I$, we get by contraposition that $I \models \Delta$, hence I is an interpolant for Γ and Δ

For the other direction, let Γ and Δ be sets of first-order formulas such that $\Gamma \cup \Delta$ is unsatisfiable. Then $\Gamma \models \neg\Delta$, hence by Theorem 1.2, there exists an interpolant I of Γ and $\neg\Delta$. But as thus $I \models \neg\Delta$, we get by contraposition that $\Delta \models \neg I$, so I is a reverse interpolant for Γ and Δ. □

As the notions of interpolation and reverse interpolation in this sense coincide, we will in the following only speak of interpolation where will be clear from the context which definition applies.

Lemma 1.5. *Let $\Gamma, \Gamma', \Delta, \Delta'$ be sets of first order formulas such that $\Gamma \Leftrightarrow \Gamma'$ and $\Delta \Leftrightarrow \Delta'$ and $\mathrm{L}(\Gamma) \cap \mathrm{L}(\Delta) = \mathrm{L}(\Gamma') \cap \mathrm{L}(\Delta')$. Then I is an interpolant for Γ and Δ if and only if I is an interpolant for Γ' and Δ'.*

Proof. Clearly $\Gamma \models I$ holds if and only if $\Gamma' \models I$ and similarly $\Delta \models \neg I$ holds if and only if $\Delta' \models \neg I$. As the intersections of the respective languages coincide, the language condition on I is satisfied in both directions. □

Remark. In Lemma 1.5, it is not sufficient to require that $\Gamma \Leftrightarrow \Gamma'$ and $\Delta \Leftrightarrow \Delta'$. Consider the example where $\Gamma = \Delta = \{\forall x(x = c)\}$ and $\Gamma' = \Delta' = \{\forall x(x = d)\}$. Then even though Γ and Γ' as well as Δ and Δ' have the same models, $\mathrm{L}(\Gamma) \cap \mathrm{L}(\Delta) = \{c\}$ whereas $\mathrm{L}(\Gamma') \cap \mathrm{L}(\Delta') = \{d\}$. Therefore $\forall x(x = c)$ is an interpolant for Γ and Δ but not for Γ' and Δ'. △

In the context of interpolation, every non-logical symbol is assigned a color which indicates its origin(s).

Definition 1.6 (Coloring)**.** A non-logical symbol is said to be Γ *(Δ)-colored* if it only occurs in Γ (Δ) and *grey* in case it occurs in both Γ and Δ. A symbol is *colored* if it is Γ- or Δ-colored. A term is a *Φ-term* if its outermost symbol is Φ-colored.

A term t is *mixed-colored* if t is Φ-colored for some Φ and t contains a term which is colored but not Φ-colored.

An occurrence of a variable x is called *Φ-colored* if it is contained in a maximal Φ-colored term. It is called *colored* if it is of any color and *grey* otherwise.

△

CHAPTER 2

Calculi

In this chapter, we introduce the calculi that are used subsequently. These are resolution and sequent calculus.

2.1 Resolution

Resolution calculus, in the formulation as given here, is a sound and complete calculus for first-order logic with equality. Due to the simplicity of its rules, it is widely used in the area of automated deduction.

Definition 2.1. A *clause* is a finite set of literals. The empty clause will be denoted by $\square$. A *resolution refutation* of a set of clauses Γ is a derivation of $\square$ consisting of applications of resolution rules (cf. Figure 2.1) starting from clauses in Γ. $\triangle$

Clauses will usually be denoted by C or D, literals by l or l' and positions by p.

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

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

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

Figure 2.1: The rules of resolution calculus

Theorem 2.2. *A clause set Γ is unsatisfiable if and only if there is resolution refutation of Γ.*

Proof. See [Rob65]. $\square$

Definition 2.3 (Tree refutations)**.** A resolution refutation is a *tree refutation* if every clause is used at most once. △

The following lemma shows that the restriction to tree refutations does not restrict the calculus given that we allow multisets as initial clause sets.

Lemma 2.4. *Every resolution refutation can be transformed into a tree refutation.*

Proof. Let π be a resolution refutation of a set of clauses Φ. We show that π can be transformed into a tree refutation by induction on the number of clauses that are used multiple times.

Suppose that no clause is used more than once in π. Then π is a tree refutation.

Otherwise let Ψ be the set of clauses which is used multiple times. Let $C \in \Psi$ be such that no clause $D \in \Psi$ is used in the derivation leading to C. Let χ be the derivation leading to C.

Suppose C is used m times. We create another resolution refutation π' from π which contains m copies of χ and replaces the ith use of the clause C by the final clause of the ith copy of χ, $1 \leqslant i \leqslant m$. In order to ensure that the sets of variables of the input clauses are disjoint, we rename the variables in each copy of χ and adapt π' accordingly. Hence π' is a resolution refutation of Φ where $m - 1$ clauses are used more than once. □

2.2 Resolution and Interpolation

In order to apply resolution to arbitrary first-order formulas, they have to be converted to clauses first. This usually makes use of intermediate normal forms which are defined as follows:

Definition 2.5. A formula is in *Negation Normal Form (NNF)* if negations only occur directly before of atoms. A formula is in *Conjunctive Normal Form (CNF)* if it is a conjunction of disjunctions of literals. △

In this context, the conjuncts of a CNF-formula are interpreted as clauses. A well-established procedure for the translation to CNF is comprised of the following steps:

1. NNF-Transformation
2. Skolemisation
3. CNF-Transformation

Step 1 can be achieved by solely pushing the negation inwards. As this transformation yields logically equivalent formulas without affecting the language, by Lemma 1.5, the set of interpolants remains unchanged. Step 2 and 3 on the other hand do not produce logically equivalent formulas since they introduce new symbols. In this section, we will show that they nonetheless do preserve the set of interpolants. This fact is vital for the use of resolution-based methods for interpolant computation of arbitrary formulas.

2.2.1 Interpolation and Skolemisation

Skolemisation is a procedure for replacing existential quantifiers by Skolem terms:

Definition 2.6. Let $V_{\exists x}$ be the set of universally bound variables whose scope includes the occurrence of $\exists x$ in a formula. The skolemisation of a formula A in NNF, denoted by $\mathrm{sk}(A)$, is the result of replacing every occurrence of an existential quantifier $\exists x$ in A by a term $f(y_1, \ldots, y_n)$ where f is a new Skolem function symbol and $V_{\exists x} = \{y_1, \ldots, y_n\}$. In case $V_{\exists x}$ is empty, the occurrence of $\exists x$ is replaced by a new Skolem constant symbol c.

The skolemisation of a set of formulas Φ is defined to be $\mathrm{sk}(\Phi) = \{\mathrm{sk}(A) \mid A \in \Phi\}$. $\triangle$

Note that due to the introduction of Skolem symbols, it is not the case that $\Phi \Leftrightarrow \mathrm{sk}(\Phi)$.

Proposition 2.7. *Let $\Gamma \cup \Delta$ be unsatisfiable. Then I is an interpolant for $\Gamma \cup \Delta$ if and only if it is an interpolant for $\mathrm{sk}(\Gamma) \cup \mathrm{sk}(\Delta)$.*

Proof. Since $\mathrm{sk}(\cdot)$ adds fresh symbols to both Γ and Δ individually, none of them are contained in $\mathrm{L}(\mathrm{sk}(\Gamma)) \cap \mathrm{L}(\mathrm{sk}(\Delta))$. Therefore the language condition on the interpolant is satisfied in both directions.

We conclude the proof by showing that $\Phi \models A$ iff $\mathrm{sk}(\Phi) \models A$ for $\Phi \in \{\Gamma, \Delta\}$ and $A \in \{I, \neg I\}$.

Suppose that for a model that $M \models \mathrm{sk}(\Phi)$ and $\Phi \models A$. Note that the interpretation of the skolem symbols of $\mathrm{sk}(\Phi)$ in M presents witnesses for the corresponding existential quantifiers in Φ. Hence $M \models \Phi$ and consequently $M \models A$.

On the other hand, suppose that $M \models \Phi$ and $\mathrm{sk}(\Phi) \models A$. We assume that $\mathrm{sk}(\Phi)$ only uses Skolem terms which are fresh with respect to M. Then we can extend M to a model M' of $\mathrm{sk}(\phi)$ by encoding the witness terms for the existential quantifiers in Φ in the Skolem terms of $\mathrm{sk}(\Phi)$ in M'. Then $M' \models \mathrm{sk}(\Phi)$ and thus $M' \models A$. But as $\mathrm{L}(A) \subseteq \mathrm{L}(M) \subseteq \mathrm{L}(M')$, M and M' agree on the interpretation of A, hence $M \models A$. $\square$

2.2.2 Interpolation and structure-preserving Normal Form Transformation

In the following, we describe a common method for transforming a formula A without existential quantifiers into CNF while preserving its structure. Note that the restriction to formulas without existential quantifiers can easily be established for arbitrary formulas by means of skolemisation and therefore does not limit the applicability of this procedure.

In the following, we use the notational convention that $\{\bar{y}\} \cup \{\bar{z}\} = \{\bar{x}\}$ expressing the intuition that the free variables $\bar{x}$ of a formula B are comprised of the not necessarily disjoint free variables $\bar{y}$ and $\bar{z}$ of B's direct subformulas.

Definition 2.8. For every occurrence of a subformula B of a formula A without existential quantifiers, introduce a new atom $L_B(\bar{x})$, where $\bar{x}$ are the free variables occurring in B. This atom acts as a label for the subformula. For each of them, create a defining clause D_B:

If B is atomic:

$$D_B \equiv \forall\bar{x}\big(\neg B \vee L_B(\bar{x})\big) \wedge \forall\bar{x}\big(B \vee \neg L_B(\bar{x})\big)$$

If B is of the form $\neg G$:

$$D_B \equiv \forall\bar{x}\big(L_B(\bar{x}) \vee L_G(\bar{x})\big) \wedge \forall\bar{x}\big(\neg L_B(\bar{x}) \vee \neg L_G(\bar{x})\big)$$

If B is of the form $G \wedge H$:

$$D_B \equiv \forall\bar{x}\big(\neg L_B(\bar{x}) \vee L_G(\bar{y})\big) \wedge \forall\bar{x}\big(\neg L_B(\bar{x}) \vee L_H(\bar{z})\big) \wedge \forall\bar{x}\big(L_B(\bar{x}) \vee \neg L_G(\bar{y}) \vee \neg L_H(\bar{z})\big)$$

If B is of the form $G \vee H$:

$$D_B \equiv \forall\bar{x}\big(L_B(\bar{x}) \vee \neg L_G(\bar{y})\big) \wedge \forall\bar{x}\big(L_B(\bar{x}) \vee \neg L_H(\bar{z})\big) \wedge \forall\bar{x}\big(\neg L_B(\bar{x}) \vee L_G(\bar{y}) \vee L_H(\bar{z})\big)$$

If B is of the form $G \supset H$:

$$D_B \equiv \forall\bar{x}\big(L_B(\bar{x}) \vee L_G(\bar{y})\big) \wedge \forall\bar{x}\big(L_B(\bar{x}) \vee \neg L_H(\bar{z})\big) \wedge \forall\bar{x}\big(\neg L_B(\bar{x}) \vee \neg L_G(\bar{y}) \vee L_H(\bar{z})\big)$$

If B is of the form $\forall x G$:

$$D_B \equiv \forall\bar{x}\forall x\big(\neg L_B(\bar{x}) \vee L_G(\bar{x}, x)\big) \wedge \forall\bar{x}\forall x\big(L_B(\bar{x}) \vee \neg L_G(\bar{x}, x)\big)$$

Let $D_{\Sigma(A)}$ be defined as $\bigwedge_{B\in\Sigma(A)} D_B$ and $\delta(A)$ as $D_{\Sigma(A)} \wedge \forall\bar{x} L_A(\bar{x})$, where $\Sigma(A)$ denotes the set of occurrences of subformulas of A. For a set of formulas without existential quantifiers Φ, let $\delta(\Phi) = \{\delta(B) \mid B \in \Phi\}$. △

Note that each of the D_B is in CNF, hence also $\delta(A)$ for any formula A without existential quantifiers. We continue by working out the logical relations of formulas and their image under A:

Lemma 2.9. *Let M be a model of $\delta(A)$ for a formula A without existential quantifiers. Then $M \models A$.*

Proof. We show that $M \models B \leftrightarrow L_B(\bar{x})$ for $B \in \Sigma(A)$. As $M \models \delta(A)$ directly implies that $M \models L_A$, this proves the lemma. Note that also $M \models D_{\Sigma(A)}$.

The proof is by induction on the structure of B. For the base case, let B be an atom. Then $D_B \equiv \forall\bar{x}\big(\neg B \vee L_B(\bar{x})\big) \wedge \forall\bar{x}\big(B \vee \neg L_B(\bar{x})\big)$, which due to $M \models D_B$ immediately yields $M \models B \leftrightarrow L_B(\bar{x})$.

For the induction step, we illustrate a few cases as the remaining ones are similar.

- Suppose B is of the form $\neg G$. Then:

  $$D_B \equiv \forall\bar{x}\big(L_B(\bar{x}) \vee L_G(\bar{x})\big) \wedge \forall\bar{x}\big(\neg L_B(\bar{x}) \vee \neg L_G(\bar{x})\big)$$

  By the induction hypothesis, $M \models G \leftrightarrow L_G(\bar{x})$. As $M \models D_B$, it follows that $M \models \neg L_G(\bar{x}) \leftrightarrow L_B(\bar{x})$, so $M \models \neg G \leftrightarrow L_B(\bar{x})$ and $M \models B \leftrightarrow L_B(\bar{x})$.

- Suppose B is of the form $G \vee H$. Then:

  $$D_B \equiv \forall\bar{x}\big(L_B(\bar{x}) \vee \neg L_G(\bar{y})\big) \wedge \forall\bar{x}\big(L_B(\bar{x}) \vee \neg L_H(\bar{z})\big) \wedge \forall\bar{x}\big(\neg L_B(\bar{x}) \vee L_G(\bar{y}) \vee L_H(\bar{z})\big)$$

  We can assume by the induction hypothesis that $M \models G \leftrightarrow L_G(\bar{x})$ as well as $M \models H \leftrightarrow L_H(\bar{x})$. As $M \models D_B$, we get that $M \models L_G(\bar{y}) \supset L_B(\bar{x})$, $M \models L_H(\bar{z}) \supset L_B(\bar{x})$ and $M \models L_B(\bar{x}) \supset (L_G(\bar{y}) \vee L_H(\bar{z}))$. Therefore $M \models L_B(\bar{x}) \leftrightarrow (G \vee H)$ and consequently $M \models L_B(\bar{x}) \leftrightarrow B$.

- Suppose B is of the form $\forall x G$. Then:

  $$D_B \equiv \forall\bar{x}\forall x\big(\neg L_B(\bar{x}) \vee L_G(\bar{x}, x)\big) \wedge \forall\bar{x}\forall x\big(L_B(\bar{x}) \vee \neg L_G(\bar{x}, x)\big)$$

  By the induction hypothesis, $M \models G \leftrightarrow L_G(\bar{x}, x)$. Since $M \models D_B$ and as x does not occur in $L_B(\bar{x})$, $M \models L_B(\bar{x}) \leftrightarrow \forall x G$, which is nothing else than $M \models L_B(\bar{x}) \leftrightarrow B$. □

Lemma 2.10. *Let A be a formula without existential quantifiers and M_A a model in the language* $\mathrm{L}(A)$. *Extend M_A to a model M'_A in the language* $\mathrm{L}(\delta(A))$ *such that for $B \in \Sigma(A)$, $M_A \models L_B(\bar{x})$ if and only if $M_A \models B$. Then $M'_A \models D_{\Sigma(A)}$.*

Proof. We proceed by induction on the structure of A. For the base case, suppose that A is an atom. Then $D_{\Sigma(A)} = D_A = \forall\bar{x}\big(\neg A \vee L_A(\bar{x})\big) \wedge \forall\bar{x}\big(A \vee \neg L_A(\bar{x})\big)$. Consider the case that $M'_A \models A$. Then by construction of M'_A, $M'_A \models L_A(\bar{x})$, hence D_A holds. In the case where $M'_A \models A$, we know that $M'_A \models L_A$, so D_A holds as well.

For the induction step, consider the following cases. The remaining cases can be argued analoguously.

- A is of the form $G \supset H$. Then $D_{\Sigma(A)} = D_{\Sigma(G)} \wedge D_{\Sigma(H)} \wedge D_A$. By the induction hypothesis, we get that $M'_A \models D_{\Sigma(G)}$ as well as $M'_A \models D_{\Sigma(H)}$. It remains to show that $M'_A \models D_A$, i.e. $M'_A \models \forall\bar{x}\big(L_A(\bar{x}) \vee L_G(\bar{y})\big) \wedge \forall\bar{x}\big(L_A(\bar{x}) \vee \neg L_H(\bar{z})\big) \wedge \forall\bar{x}\big(\neg L_A(\bar{x}) \vee \neg L_G(\bar{y}) \vee L_H(\bar{z})\big)$.

  Suppose that $M'_A \models A$. Then $M'_A \models G$ or $M'_A \models H$. By construction of M'_A, we furthermore have that $M'_A \models L_B(\bar{x})$ and $M'_A \models \neg L_G(\bar{y}) \vee L_H(\bar{z})$.

  Otherwise we have that $M'_A \models A$, so $M'_A \models G$ and $M'_A \models H$. Hence $M'_A \models \neg L_A(\bar{x})$, $M'_A \models L_G(\bar{y})$ and $M'_A \models L_H(\bar{z})$.

- A is of the form $\forall x B$. Then $D_{\Sigma(A)} = D_{\Sigma(B)} \wedge D_A$. By the induction hypothesis, $M'_A \models D_{\Sigma(B)}$, and we conclude by showing that $M'_A \models \forall\bar{x}\forall x\big(\neg L_A(\bar{x}) \vee L_B(\bar{x}, x)\big) \wedge \forall\bar{x}\forall x\big(L_A(\bar{x}) \vee \neg L_B(\bar{x}, x)\big)$:

  Suppose $M'_A \models A$. Then consequently, $M'_A \models \forall x B$, so $M'_A \models L_A(\bar{x})$ and $M'_A \models L_B(\bar{x}, x)$. Otherwise $M'_A \models A$. In this case $M'_A \models \forall x B$, so $M'_A \models L_A(\bar{x})$ and $M'_A \models L_B(\bar{x}, x)$. □

Lemma 2.11. *Let A be a formula and Φ a set of formulas without existential quantifiers with such that* $\mathrm{L}(A) \subseteq \mathrm{L}(\Phi)$. *Then $\Phi \models A$ if and only if $\delta(\Phi) \models A$.*

Proof. If $\Phi \models A$, then $\Phi \cup \{A\}$ is unsatisfiable and thus by the Compactness Theorem, there exists a finite $\Phi' \subseteq \Phi$ such that $\Phi' \cup \{A\}$ is unsatisfiable, or in other words $\Phi' \models A$. Extend Φ' such that $\mathrm{L}(A) \subseteq \mathrm{L}(\Phi')$. Let $B = \bigwedge_{C \in \Phi'} C$. We show that $B \models A$ if and only if $\delta(B) \models A$ by induction on the structure of B.

For the if-direction, assume that $\delta(B) \models A$ and let M be a model such that the $\mathrm{L}(B)$-reduct of M, $M|_{\mathrm{L}(B)}$, is a model of B. Let M' extend $M|_{\mathrm{L}(B)}$ as in Lemma 2.10 and hence by that lemma, $M' \models D_{\Sigma(B)}$. By the construction of M', $M' \models L_B(\bar{x})$, therefore $M' \models \delta(B)$, so by the induction hypothesis $M' \models A$. As $\mathrm{L}(A) \subseteq \mathrm{L}(B)$ and $M'|_{\mathrm{L}(B)} = M|_{\mathrm{L}(B)}$, $M \models A$.

For the only if-direction, assume that $B \models A$ and let M be a model such that $M \models \delta(B)$. By Lemma 2.9, $M \models B$ and hence $M \models A$. □

Proposition 2.12. *Let $\Gamma \cup \Delta$ be unsatisfiable and contain no existential quantifiers. Then I is an interpolant for $\Gamma \cup \Delta$ if and only if I is an interpolant for $\delta(\Gamma) \cup \delta(\Delta)$.*

Proof. As δ introduces fresh symbols for each Γ and Δ, they do not occur in any interpolant for Γ and Δ. This establishes the language condition in both directions.

Furthermore, Lemma 2.11 is applicable to interpolants I for $\Gamma \cup \Delta$ due to the language condition and demonstrates that $\Gamma \models I$ if and only if $\delta(\Gamma) \models I$ as well as $\Delta \models \neg I$ if and only if $\delta(\Gamma) \models \neg I$, which gives the result. □

At this point, we can summarize the results which enable the use of resolution based methods for calculating interpolants:

Theorem 2.13. *Let $\Gamma \cup \Delta$ be unsatisfiable. Then I is an interpolant for $\Gamma \cup \Delta$ if and only if I is an interpolant for $\delta(\mathrm{sk}(\Gamma)) \cup \delta(\mathrm{sk}(\Delta))$.*

Proof. Immediate by Proposition 2.12 and Proposition 2.7. □

2.3 Sequent Calculus

The famous sequent calculus was introduced in [Gen35]. Its use of sequents in lieu of plain formulas allows for a natural mapping of the logical relations expressed by the connectives to the structure of proofs.

Definition 2.14. For multisets of first-order formulas Γ and Δ, $\Gamma \vdash \Delta$ is called a *sequent*. In this context Γ forms the *antecedent*, whereas Δ is referred to as *succedent*.

A sequent $\Gamma \vdash \Delta$ is called *provable* if there is a sequent calculus proof of $\Gamma \vdash \Delta$. △

The rules of sequent calculus are as follows:

Axioms

$$A \vdash A \qquad\qquad \vdash t = t$$

Cut

$$\frac{\Gamma \vdash \Delta, A \qquad A, \Sigma \vdash \Pi}{\Gamma, \Sigma \vdash \Delta, \Pi}$$

Structural rules

- Contraction

$$\frac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}\,\mathrm{c} : l \qquad\qquad \frac{\Gamma \vdash \Delta, A, A}{\Gamma \vdash \Delta, A}\,\mathrm{c} : r$$

- Weakening

$$\frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}\,\mathrm{w} : l \qquad\qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, A}\,\mathrm{w} : r$$

Propositional rules

- Negation

$$\frac{\Gamma \vdash \Delta, A}{\neg A, \Gamma \vdash \Delta}\,\neg : l \qquad\qquad \frac{A, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \neg A}\,\neg : r$$

- Conjunction

$$\frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}\,\wedge : l \qquad\qquad \frac{\Gamma \vdash \Delta, A \qquad \Sigma \vdash \Pi, B}{\Gamma, \Sigma \vdash \Delta, \Pi, A \wedge B}\,\wedge : r$$

- Disjunction

$$\frac{\Gamma, A \vdash \Delta \qquad \Sigma, B \vdash \Pi}{\Gamma, \Sigma, A \vee B \vdash \Delta, \Pi}\,\vee : l \qquad\qquad \frac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, A \vee B}\,\vee : r$$

- Implication

$$\frac{\Gamma \vdash A, \Delta \qquad \Sigma, B \vdash \Pi}{\Gamma, \Sigma, A \supset B \vdash \Delta, \Pi}\,\supset : l \qquad\qquad \frac{\Gamma, A \vdash \Delta, B}{\Gamma \vdash \Delta, A \supset B}\,\supset : r$$

Quantifier rules

- Universal

$$\frac{\Gamma, A[x/t] \vdash \Delta}{\Gamma, \forall x A \vdash \Delta} \forall : l \qquad \frac{\Gamma \vdash \Delta, A[x/y]}{\Gamma \vdash \Delta, \forall x A} \forall : r$$

- Existential

$$\frac{\Gamma, A[x/y] \vdash \Delta}{\Gamma, \exists x A \vdash \Delta} \exists : l \qquad \frac{\Gamma \vdash \Delta, A[x/t]}{\Gamma \vdash \Delta, \exists x A} \exists : r$$

(provided no free variable of t becomes bound in $A[x/t]$ and y does not occur free in Γ, Δ or A)

Equality rules

- Left rules

$$\frac{\Gamma, A[t]_p \vdash \Delta \qquad \Sigma \vdash \Pi, s = t}{\Gamma, \Sigma, A[s]_p \vdash \Delta, \Pi} = : l_1 \qquad \frac{\Gamma, A[s]_p \vdash \Delta \qquad \Sigma \vdash \Pi, s = t}{\Gamma, \Sigma, A[t]_p \vdash \Delta, \Pi} = : l_2$$

- Right rules

$$\frac{\Gamma \vdash \Delta, A[t]_p \qquad \Sigma \vdash \Pi, s = t}{\Gamma, \Sigma \vdash \Delta, \Pi, A[s]_p} = : r_1 \qquad \frac{\Gamma \vdash \Delta, A[s]_p \qquad \Sigma \vdash \Pi, s = t}{\Gamma, \Sigma \vdash \Delta, \Pi, A[t]_p} = : r_2$$

(provided no free variable of s or t becomes bound in $A[t]_p$ or $A[s]_p$)

Figure 2.2: The rules of sequent calculus

For the purposes of this thesis, we usually consider the cut-free fragment of sequent calculus.

Theorem 2.15. *Cut-free sequent calculus is sound and complete.*

Chapter 3

Reduction to First-Order Logic without Equality

A common theme of proofs is to avoid the tedious effort of proving the result from first principles by reducing the problem to one that is easier to solve. In this instance, we are able to give a reduction for finding interpolants in first-order logic *with* equality to first-order logic *without* equality, where it is simpler to give an appropriate algorithm. This method is due to Craig ([Cra57b]).

In order to simplify notation, we shall consider constant symbols to be function symbols of arity 0 in this section. The general layout of this approach is the following: From two sets Γ and Δ, where $\Gamma \cup \Delta$ is unsatisfiable, we compute two sets Γ' and Δ' which do not make use of equality but simulate the effects of equality in Γ and Δ via axioms. In the process of this transformation, also function symbols are replaced by predicate symbols with appropriate axioms to make sure that the behaviour of these function-representing predicates is compatible to the one of actual functions. Now an interpolant for Γ' and Δ' can be derived using an algorithm that is only capable of handling predicate symbols as all other non-logical symbols have been removed. Since the additional axioms ensure that the newly added predicate symbols mimic equality and functions respectively, we will see that the occurrences of these predicates in the interpolant can be translated back to occurrences of equality and function symbols in first-order logic with equality in the language of Γ and Δ, thereby yielding the originally desired interpolant.

3.1 Translation of formulas

As we shall see in this section, first-order formulas with equality can be transformed into first-order formulas without equality in a way that is satisfiability-preserving, which is sufficient for our purposes.

First, we define axioms in a language with fresh symbols which allows for simulation of equality and functions in first order logic without equality and function symbols:

Definition 3.1 (Translation of languages)**.** For a first-order language $\mathcal{L}$ and fresh predicate symbols E and F_f for $f \in \mathrm{FS}(\mathcal{L})$, $\mathrm{T}(\mathcal{L})$ denotes $(\mathcal{L} \cup \{E\} \cup \{F_f \mid f \in \mathrm{FS}(\mathcal{L})\}) \setminus (\{=\} \cup \mathrm{FS}(\mathcal{L}))$. △

Definition 3.2 (Equality and function axioms)**.** For a first-order language $\mathcal{L}$ we define the following axions in $\mathrm{T}(\mathcal{L})$:

$$\mathrm{F_{Ax}}(\mathcal{L}) \stackrel{\mathrm{def}}{=} \bigcup_{f \in \mathrm{FS}(\mathcal{L})} \forall \bar{x} \exists y (F_f(\bar{x}, y) \wedge (\forall z (F_f(\bar{x}, z) \supset E(y, z))))$$

$$\mathrm{Refl}(P) \stackrel{\mathrm{def}}{=} \forall x P(x, x)$$

$$\mathrm{Congr}(P) \stackrel{\mathrm{def}}{=} \forall x_1 \forall y_1 \ldots \forall x_{\mathrm{ar}(P)} \forall y_{\mathrm{ar}(P)} ((E(x_1, y_1) \wedge \ldots \wedge E(x_{\mathrm{ar}(P)}, y_{\mathrm{ar}(P)})) \supset (P(x_1, \ldots, x_{\mathrm{ar}(P)}) \supset P(y_1, \ldots, y_{\mathrm{ar}(P)})))$$

$$\mathrm{E_{Ax}}(\mathcal{L}) \stackrel{\mathrm{def}}{=} \mathrm{Refl}(E) \cup \bigcup_{\substack{P \in \mathrm{PS}(\mathcal{L}) \cup \{E\} \cup \\ \{F_f \mid f \in \mathrm{FS}(\mathcal{L})\}}} \mathrm{Congr}(P)$$ △

$\mathrm{Refl}(P)$ will be referred to as reflexivity axiom of P, $\mathrm{Congr}(P)$ as congruence axiom of P. As any model of $\mathrm{E_{Ax}}(\mathcal{L})$ requires $\mathrm{Refl}(E)$ and $\mathrm{Congr}(E)$, E is also symmetric and transitive in the model:

Proposition 3.3. *In every model of* $\mathrm{Refl}(E)$ *and* $\mathrm{Congr}(E)$*, E is an equivalence relation.*

Proof. Let M be a model of $\mathrm{Refl}(E)$ and $\mathrm{Congr}(E)$. Then M clearly is reflexive. Due to $M \models \mathrm{Congr}(E)$, $M \models \forall x \forall y (E(x, y) \wedge E(x, x)) \supset (E(x, x) \supset E(y, x))$. As we know that E is reflexive, this simplifies to $M \models \forall x \forall y (E(x, y) \supset E(y, x))$, i.e. E is symmetric in M. We show the transitivity of E by another instance of $\mathrm{Congr}(E)$: $M \models \forall x \forall y \forall z ((E(y, x) \wedge E(y, z)) \supset (E(y, y) \supset E(x, z)))$, As E is reflexive and symmetric, we get that $M \models \forall x \forall y \forall z ((E(x, y) \wedge E(y, z)) \supset E(x, z))$. □

We continue by defining the translation procedure for formulas:

Definition 3.4 (Translation and inverse translation of formulas)**.** Let A be a first-order formula and E and F_f for $f \in \mathrm{FS}(A)$ be fresh predicate symbols. Then $\mathrm{T}(A)$ is the result of applying the following algorithm to A:

1. Replace every occurrence of $s = t$ in A by $E(s, t)$
2. As long as there is an occurrence of a function symbol f in A:
   Let B be the atom in which f occurs as outermost symbol of a term. Then B is of the form $P(s_1, \ldots, s_{j-1}, f(\bar{t}), s_{j+1}, \ldots s_m)$. Replace B in A by $\exists y (F_f(\bar{t}, y) \wedge P(s_1, \ldots, s_{j-1}, y, s_{j+1}, \ldots s_m))$ for a fresh variable y.

Moreover, let the inverse operation $\mathrm{T}^{-1}(B)$ for formulas B in the language $\mathrm{T}(L(A))$ be defined as the result of applying the following algorithm to B:

1. Replace every occurrence of $E(s, t)$ in B by $s = t$.
2. For every $f \in FS(A)$, replace every occurrence of $\exists y (F_f(\bar{t}, y) \wedge P(s_1, \ldots, s_{j-1}, y, s_{j+1}, \ldots s_m))$ in B by $P(s_1, \ldots, s_{j-1}, f(\bar{t}), s_{j+1}, \ldots s_m)$

3. For every $f \in FS(A)$, replace every occurrence of $F_f(\bar{t}, s)$ by $f(\bar{t}) = s$.

For sets of first-order formulas Φ, let $\mathrm{T}(\Phi) \stackrel{\text{def}}{=} \bigcup_{A\in\Phi} \mathrm{T}(A)$ and $\mathrm{T}^{-1}(\Phi) \stackrel{\text{def}}{=} \bigcup_{A\in\Phi} \mathrm{T}^{-1}(A)$. △

Remark. Let $\mathcal{L}$ be a language. Step 2 and 3 of T^{-1} are both concerned with replacing occurrences of F_f by occurrences of f for $f \in \mathrm{FS}(\mathcal{L})$, but are relevant in different contexts.

Step 2 of T^{-1} is the precise inverse of step 2 of T in the sense that for any formula A, $\mathrm{T}^{-1}(\mathrm{T}(A)) = A$ as we will show in Lemma 3.5. In this context, step 3 has no effect, as all occurrences of F_f have been introduced by $\mathrm{T}(\cdot)$ and are consequently of exactly the form that is handled by step 2. So the algorithm is in this regard complete even without step 3.

On the other hand, if arbitrary formulas in the language $\mathrm{T}(\mathcal{L})$ are given, they in general do not match that pattern and are only translated to $\mathcal{L}$ in step 3. Note that T^{-1} without step 2 yields a complete algorithm, as any formula that is handled there can also be processed in step 3. In such a procedure, $\mathrm{T}^{-1}(\mathrm{T}(A))$ and A are in general not syntactically equal for formulas A but only logically equivalent. △

Lemma 3.5. *Let A be a first-order formula and Φ be a set of first-order formulas. Then $\mathrm{T}^{-1}(\mathrm{T}(A)) = A$ and $\mathrm{T}^{-1}(\mathrm{T}(\Phi)) = \Phi$.*

Proof. Step 1 and 2 in the algorithms T and T^{-1} are each concerned with a different set of symbols and therefore do not interfere with each other. Moreover, the respective steps in both algorithms are the inverse of each other. For step 1, this is immediate and for step 2, consider that all occurrences of F_f for $f \in \mathrm{FS}(A)$ in $\mathrm{T}(A)$ have been introduced by T and are consequently of the form $\exists y(F_f(\bar{t}, y) \wedge P(s_1, \ldots, s_{j-1}, y, s_{j+1}, \ldots s_m))$, which is replaced by $P(s_1, \ldots, s_{j-1}, f(\bar{t}), s_{j+1}, \ldots s_m)$ by T^{-1}. As no occurrences of F_f remain, step 3 of T^{-1} leaves the formula unchanged. □

Definition 3.6 (Translation of formulas including axioms)**.** For first-order formulas A, let $\mathrm{T}_{\mathrm{Ax}}(A) \stackrel{\text{def}}{=} \left(\bigwedge_{B\in\mathrm{F}_{\mathrm{Ax}}(\mathrm{L}(A))} B\right) \wedge \left(\bigwedge_{B\in\mathrm{E}_{\mathrm{Ax}}(\mathrm{L}(A))} B\right) \wedge \mathrm{T}(A)$ and for sets of first-order formulas Φ, let $\mathrm{T}_{\mathrm{Ax}}(\Phi) \stackrel{\text{def}}{=} \mathrm{F}_{\mathrm{Ax}}(\mathrm{L}(\Phi)) \cup \mathrm{E}_{\mathrm{Ax}}(\mathrm{L}(\Phi)) \cup \mathrm{T}(\Phi)$. △

Note that $\mathrm{T}_{\mathrm{Ax}}(A)$ contains neither the equality predicate nor function symbols but additional predicate symbols instead. More formally:

Lemma 3.7.

1. *Let Φ be a set of first-order formulas. Then $\mathrm{T}_{\mathrm{Ax}}(\Phi)$ is in the language $\mathrm{T}(\mathrm{L}(\Phi))$.*
2. *If Ψ is in the language $\mathrm{T}(\mathcal{L})$, then $\mathrm{T}^{-1}(\Psi)$ is in the language $\mathcal{L}$.*

Proposition 3.8. *Let Φ be a set of first-order formulas.*

1. *If Φ is satisfiable, then so is $\mathrm{T}_{\mathrm{Ax}}(\Phi)$.*
2. *Let $\mathcal{L}$ be a first-order language and Φ a set of first-order formulas in the language $\mathrm{T}(\mathcal{L})$. If $\mathrm{F}_{\mathrm{Ax}}(\mathcal{L}) \cup \mathrm{E}_{\mathrm{Ax}}(\mathcal{L}) \cup \Phi$ is satisfiable, then so is $\mathrm{T}^{-1}(\Phi)$.*

Proof. Suppose Φ is satisfiable. Let M be a model of Φ. We show that $\mathrm{T}_{\mathrm{Ax}}(\Phi)$ is satisfiable by extending M to the language $\mathrm{L}(\Phi) \cup \{E\} \cup \{F_f \mid f \in \mathrm{FS}(A)\}$ and proving that the extended model satisfies $\mathrm{T}_{\mathrm{Ax}}(\Phi)$.

First, let $M \models E(s,t)$ if and only if $M \models s = t$. By reflexivity of equality, it follows that $M \models \mathrm{Refl}(E)$. As any predicate, in particular E and F_f for every $f \in \mathrm{FS}(\Phi)$, satisfy the congruence axiom with respect to $=$, by the definition of E in M, they satisfy the congruence axiom with respect to E. Therefore M is a model of $\mathrm{E}_{\mathrm{Ax}}(\mathrm{L}(\Phi))$.

Second, let $M \models F_f(\bar{x}, y)$ if and only if $M \models f(\bar{x}) = y$ for all $f \in \mathrm{FS}(\Phi)$. Since M is a model of Φ, it maps every function symbol f to a function, which by definition returns a unique result for every combination of parameters. This however is precisely the logical requirement on F_f stated by $\mathrm{F}_{\mathrm{Ax}}(\mathrm{L}(\Phi))$, hence M is a model of $\mathrm{F}_{\mathrm{Ax}}(\mathrm{L}(\Phi))$.

Lastly, we show that $M \models \mathrm{T}(A)$ for all $A \in \Phi$. By the above definition of E in M, step 1 of the algorithm in Definition 3.4 yields a formula that is satisfied by M as it satisfies every formula of Φ. For step 2, suppose $P(s_1, \ldots, s_{j-1}, f(\bar{t}), s_{j+1}, \ldots s_m)$ does (not) hold under M. Let y be such that $M \models f(\bar{t}) = y$. By our definition of F_f under M, $M \models F_f(\bar{t}, y)$ with this unique y. Hence $\exists y(F_f(\bar{t}, y) \wedge P(s_1, \ldots, s_{j-1}, y, s_{j+1}, \ldots s_m))$ does (not) hold under M.

For 2, suppose $\mathrm{F}_{\mathrm{Ax}}(\mathcal{L}) \cup \mathrm{E}_{\mathrm{Ax}}(\mathcal{L}) \cup \Phi$ is satisfiable and let M be a model of it.

First, note that as $M \models \mathrm{E}_{\mathrm{Ax}}(\mathcal{L})$, by Proposition 3.3, E^M is an equivalence relation. Let D be the domain of M. We build a model M' with the domain $D' = D/E^M$, i.e. the the congruence relation of D modulo E^M. The interpretation $\mathcal{I}'$ of M' is obtained from the interpretation $\mathcal{I}$ of M by replacing every occurrence of a domain element d by its respective congruence class $\{x \mid (d,x) \in E^M\}$. As $M \models \mathrm{E}_{\mathrm{Ax}}(\mathcal{L})$, it satisfies the congruence axioms with respect to every function and predicate symbol, and therefore $\mathcal{I}'$ is well-defined. Due to this construction, $M' \models s = t$ if and only if $M \models E(s,t)$ for all terms s and t.

how to write this in a nicer way?

Second, let $M \models f(\bar{t}) = s$ if and only if $M \models F_f(\bar{t}, s)$ for all $f \in \mathrm{FS}(\mathcal{L})$. As by assumption M is a model of $\mathrm{F}_{\mathrm{Ax}}(A)$, we know that for every $\bar{t}$, some s with $M \models F(\bar{t}, s)$ exists and is uniquely defined. Hence f in M refers to a well-defined function.

Lastly, to show that $M \models \mathrm{T}^{\text{-}1}(\Phi)$, consider that the interpretations of the predicates E and $=$ coincide in M. Furthermore, let B be an occurrence of $\exists y(F_f(\bar{t}, y) \wedge P(s_1, \ldots, s_{j-1}, y, s_{j+1}, \ldots s_m))$ for some $f \in \mathrm{FS}(\mathcal{L})$ in Φ. Then by the above definition of f in M, we have that B is in M equivalent to $\exists y f(\bar{t}) = y) \wedge P(s_1, \ldots, s_{j-1}, y, s_{j+1}, \ldots s_m))$, which due to f being a function is equivalent to $M \models P(s_1, \ldots, s_{j-1}, f(\bar{t}), s_{j+1}, \ldots s_m))$.

Similarly, let B be an occurrence of $F_f(\bar{t}, s)$ in Φ. Then by our above definition of f in M, we have that $M \models f(\bar{t}) = s$ iff $M \models B$. □

Corollary 3.9. *Let Φ be a set of first-order formulas. Then Φ is satisfiable if and only if $\mathrm{T}_{\mathrm{Ax}}(\Phi)$ is satisfiable.*

Proof. The left-to-right direction is directly given in Proposition 3.8. For the other direction, consider that by Proposition 3.8, $\mathrm{T}^{\text{-}1}(\mathrm{T}(\Phi))$ is satisfiable, which by Lemma 3.5 is nothing else than Φ. □

3.2 Computation of interpolants

For the proof of the interpolation theorem by reduction we require an algorithm that operates in first-order logic without equality and function symbols, which we describe in this section.

Remark. As the idea of this reduction is to simplify the problem by amongst others not considering function symbols, resolution-based methods can not be employed in a direct manner. This is because function symbols appear naturally in them as they usually handle existential quantification by means of skolemisation, i.e. a new function symbol is introduced for every occurrence of an existential quantifier in the scope of a universal quantifier. Translating the skolemised formulas to a language without function symbols as described in Definition 3.4 is of no avail since this translation introduces new existential quantifiers for every function symbol it encounters, necessitating skolemisation yet again. △

Lemma 3.10. *Let Γ and Δ be sets of first-order formulas such that the equality symbol does not occur in them and $\Gamma \vdash \Delta$ is provable in sequent calculus. Then there exists a proof of $\Gamma \vdash \Delta$ that does not contain the equality symbol.*

Proof. Let π be a proof of $\Gamma \vdash \Delta$.

this is not finished

Suppose π contains an instance of the equality axiom $\vdash t = t$ for a term t. As no equality symbol is contained in the end sequent, there has to be a rule application in π which removes either $t = t$. Only instances of equality rules or cut are capable of this.

Consider the case that an equality rule removed $t = t$. As the cases for $= : l_1$, $= : l_2$, $= : r_1$ and $= : r_2$ are similar, we only consider the case of $= : l_1$. The proof π' leading up to the rule application that removes of the occurrence of $t = t$ is of the form:

$$\dfrac{\begin{matrix}\varphi \\ \Gamma, A[t]_p \vdash \Delta\end{matrix} \qquad \begin{matrix}\psi \\ \Sigma \vdash \Pi, t = t\end{matrix}}{\Gamma, \Sigma, A[t]_p \vdash \Delta, \Pi} = : l_1$$

We can replace π' in π by the following to obtain a proof without an occurrence of the equality symbol:

$$\dfrac{\dfrac{\begin{matrix}\varphi \\ \Gamma, A[t]_p \vdash \Delta\end{matrix}}{\Gamma, \Sigma, A[t]_p \vdash \Delta}\,\mathrm{w} : l}{\Gamma, \Sigma, A[t]_p \vdash \Delta, \Pi}\,\mathrm{w} : r$$

TODO: CUT

Suppose π contains an instance of the axiom $A \vdash A$ such that the equality symbol occurs in A. Then A is of the form $s = t$ for terms s and t. While the occurrence in the consequent might be eliminated by an equality rule application, due to the subformula property, there is no rule in cut-free sequent calculus such that the occurrence in the antecedent is removed. Hence it appears in the final sequent, which contradicts the assumption.

Suppose π contains an instance of $\mathrm{w} : l$ such that the equality symbolc occurs in the principal formula A. This case can be argued similarly as for occurrences of A as antecedent of an axiom $A \vdash A$.

Suppose π contains an instance of $\mathrm{w} : r$ such that the equality symbolc occurs in the principal formula A. Then as it does not occur in the end sequent, it is removed by either an instance of an equality rule or the cut rule. Suppose it is removed via an equality rule. We consider the case of $= : l_1$.

$$\dfrac{\dfrac{\dfrac{\varphi}{\Gamma \vdash \Delta}\,\mathrm{w} : r}{\Gamma \vdash \Delta, s = t} \qquad \begin{array}{c}\vdots\\ \Sigma \vdash \Pi, s = t\end{array}}{\Lambda, A[t]_p \vdash \Theta \qquad \Lambda, \Sigma, A[s]_p \vdash \Theta, \Pi} = : l_1$$

TODO
(beweis kann l"anger werden, aber wie geht das? inessential cuts bleiben doch "uber?)
□

We now show that interpolants can be computed by means of a sequent calculus based procedure by Maehara. It is slightly stronger than the required statement as it allows for interpolants of partitions of sequents:

Definition 3.11 (Partition of sequents)**.** A partition of a sequent $\Gamma \vdash \Delta$ is denoted by $\langle(\Gamma_1; \Delta_1), (\Gamma_2; \Delta_2)\rangle$, where $\Gamma_1 \uplus \Gamma_2 = \Gamma$ and $\Delta_1 \uplus \Delta_2 = \Delta$. △

Lemma 3.12 (Maehara)**.** *Let Γ and Δ be sets of first-order clauses without equality and function symbols such that $\Gamma \vdash \Delta$ is provable in cut-free 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. *$\mathrm{L}(I) \subseteq \mathrm{L}(\Gamma_1, \Delta_1) \cap \mathrm{L}(\Gamma_2, \Delta_2)$*

Proof. We prove this lemma by induction on the number of inferences in a cut-free proof of $\Gamma \vdash \Delta$. By Lemma 3.10, we can assume that no equality symbol occurs in the proof, so equality rules need not be considered.

Base case. Suppose no rules were applied. Then $C \vdash D$ is of one of the form $A \vdash A$. We give interpolants for any of the four possible partitions:

1. $\langle(A; A), (;)\rangle$: $I = \bot$
2. $\langle(;), (A; A)\rangle$: $I = \top$
3. $\langle(; A), (A;)\rangle$: $I = \neg A$
4. $\langle(A;), (; A)\rangle$: $I = A$

Structural rules. Suppose the property holds for n rule applications and the $(n+1)$th rule is a structural one.

- The last rule application is an instance of $\mathrm{c} : l$. Then it is of the form:

$$\frac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}\,\mathrm{c} : l$$

  There are two possible partition schemes: of $\Gamma, A \vdash \Delta$:

  1. $\chi = \langle(\Gamma_1, A; \Delta_1), (\Gamma_2; \Delta_2)\rangle$. By the induction hypothesis, we know that there is an interpolant I for the partition $\langle(\Gamma_1, A, A; \Delta_1), (\Gamma_2; \Delta_2)\rangle$ of the upper sequent. I serves as interpolant for χ as well.
  2. $\chi = \langle(\Gamma_1; \Delta_1), (\Gamma_2, A; \Delta_2)\rangle$. By a similar argument, we get that there is an interpolant I for $\langle(\Gamma_1; \Delta_1), (\Gamma_2, A, A; \Delta_2)\rangle$, which again is also an interpolant for χ.

  The case of $\mathrm{c} : r$ is analogous.

- The last rule application is an instance of $\mathrm{w} : r$. Then it is of the form:

$$\frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, A}\,\mathrm{w} : r$$

  By the induction hypothesis, there exists an interpolant I for any partition $\langle(\Gamma_1; \Delta_1), (\Gamma_2; \Delta_2)\rangle$ of $\Gamma \vdash \Delta$. Clearly I remains an interpolant when adding A to either Δ_1 or Δ_2.

  The case of $\mathrm{w} : l$ is analogous.

Propositional rules. Suppose the property holds for n rule applications and the $(n+1)$th rule is a propositional one.

- The last rule application is an instance of $\neg : l$. Then it is of the form:

$$\frac{\Gamma \vdash \Delta, A}{\neg A, \Gamma \vdash \Delta}\,\neg : l$$

  There are two possible partition schemes of $\Gamma, \neg A \vdash \Delta$:

  1. $\chi = \langle(\Gamma_1, \neg A; \Delta_1), (\Gamma_2; \Delta_2)\rangle$. By the induction hypothesis, there exists an interpolant I for the partition $\langle(\Gamma_1; \Delta_1, A), (\Gamma_2; \Delta_2)\rangle$ of the upper sequent. Clearly I is an interpolant for χ as well.
  2. $\chi = \langle(\Gamma_1; \Delta_1), (\Gamma_2, \neg A; \Delta_2)\rangle$. A similar argument goes through.

  The case of $\neg : r$ is analogous.

- The last rule application is an instance of $\supset : l$. Then it is of the form:

$$\frac{\Gamma \vdash \Delta, A \qquad \Sigma, B \vdash \Pi}{\Gamma, \Sigma, A \supset B \vdash \Delta, \Pi}\,\supset : l$$

  There are two possible partition schemes of $\Gamma, A \supset B \vdash \Delta$:

1. $\chi = \langle(\Gamma_1, \Sigma_1, A \supset B; \Delta_1, \Pi_1), (\Gamma_2, \Sigma_2; \Delta_2, \Pi_2)\rangle$. By the induction hypothesis, there is an interpolant I_1 for the partition $\langle(\Gamma_1; \Delta_1, A), (\Gamma_2; \Delta_2)\rangle$ of the left upper sequent. Hence for I_1, we have that $\Gamma_1 \vdash \Delta_1, A, I_1$ and $I_1, \Gamma_2 \vdash \Delta_2$ are provable.
   Moreover, we also get by the induction hypothesis that there is an interpolant I_2 for the partition $\langle(\Sigma_1, B; \Pi_1), (\Sigma_2; \Pi_2)\rangle$ of the right upper sequent. Therefore $\Sigma_1, B \vdash \Pi_1, I_2$ and $I_2, \Sigma_2 \vdash \Pi_2$ are provable.
   Using these prerequisites, we first establish that $I_1 \vee I_2$ fulfills conditions 1 and 2 of an interpolant for χ:

$$\dfrac{\dfrac{\Gamma_1 \vdash \Delta_1, A, I_1 \qquad \Sigma_1, B \vdash \Pi_1, I_2}{\Gamma_1, \Sigma_1, A \supset B \vdash \Delta_1, \Pi_1, I_1, I_2}\supset : l}{\Gamma_1, \Sigma_1, A \supset B \vdash \Delta_1, \Pi_1, I_1 \vee I_2}\vee : r$$

$$\dfrac{I_1, \Gamma_2 \vdash \Delta_2 \qquad I_2, \Sigma_2 \vdash \Pi_2}{I_1 \vee I_2, \Gamma_2, \Sigma_2 \vdash \Delta_2, \Pi_2}\vee : l$$

   To show that also condition 3 is satisfied, consider that by the induction hypothesis, it holds that:

$$\mathrm{L}(I_1) \subseteq \mathrm{L}(\Gamma_1, \Delta_1, A) \cap \mathrm{L}(\Gamma_2, \Delta_2)$$
$$\mathrm{L}(I_2) \subseteq \mathrm{L}(\Sigma_1, B, \Pi_1) \cap \mathrm{L}(\Sigma_2, \Pi_2)$$

   Therefore

$$\begin{aligned}
\mathrm{L}(I_1) \cup \mathrm{L}(I_2) &\subseteq (\mathrm{L}(\Gamma_1, \Delta_1, A) \cap \mathrm{L}(\Gamma_2, \Delta_2)) \cup (\mathrm{L}(\Sigma_1, B, \Pi_1) \cap \mathrm{L}(\Sigma_2, \Pi_2)) \\
&\Downarrow \\
\mathrm{L}(I_1) \cup \mathrm{L}(I_2) &\subseteq (\mathrm{L}(\Gamma_1, \Delta_1, A) \cup \mathrm{L}(\Sigma_1, B, \Pi_1)) \cap (\mathrm{L}(\Gamma_2, \Delta_2) \cup \mathrm{L}(\Sigma_2, \Pi_2)) \\
&\Updownarrow \\
\mathrm{L}(I_1 \vee I_2) &\subseteq \mathrm{L}(\Gamma_1, \Sigma_1, A \supset B, \Delta_1, \Pi_1) \cap \mathrm{L}(\Gamma_2, \Sigma_2, \Delta_2, \Pi_2)
\end{aligned}$$

2. $\chi = \langle(\Gamma_1, \Sigma_1; \Delta_1, \Pi_1), (\Gamma_2, \Sigma_2, A \supset B; \Delta_2, \Pi_2)\rangle$. The argument for this case is similar using $I_1 \wedge I_2$ as interpolant.

For the other binary connectives $\wedge : l$, $\wedge : r$, $\vee : l$, $\vee : r$ and $\supset : r$, similar arguments go through, where the interpolant is always either the conjunction or the disjunction of the interpolants of partitions of the preceding sequents.

Quantifier rules. Suppose the property holds for n rule applications and the $(n + 1)$th rule is a quantifier rule.

- The last rule application is an instance of $\forall : l$. Then it is of the form:

$$\dfrac{\Gamma, A[x/y] \vdash \Delta}{\Gamma, \forall x A \vdash \Delta}\forall : l$$

Note that since we have excluded function symbols to occur in the final sequent (and constant symbols are treated as function symbols of arity 0),

it is only possible to find a proof where these don't occur, but they might occur in a stupid proof. add a lemma as for equality?

There are two possible partition schemes of $\Gamma, \forall x A \vdash \Delta$:

1. $\langle(\Gamma_1, \forall x A; \Delta_1), (\Gamma_2; \Delta_2)\rangle$. By the induction hypothesis, there is an interpolant I of the partition $\langle(\Gamma_1, A[x/y]; \Delta_1), (\Gamma_2; \Delta_2)\rangle$. Hence for I, $\Gamma_1, A[x/y] \vdash \Delta_1, I$ and $I, \Gamma_2 \vdash \Delta_2$ are provable. By an application of $\forall : l$ to the first sequent we get $\Gamma_1, \forall x A \vdash \Delta_1, I$, so I satisfies conditions 1 and 2 of being an interpolant for χ.
   In order to show that also $\mathrm{L}(I) \subseteq \mathrm{L}(\Gamma_1, \forall x A, \Delta_1) \cap \mathrm{L}(\Gamma_2, \Delta_2)$, consider that by the induction hypothesis, $\mathrm{L}(I) \subseteq \mathrm{L}(\Gamma_1, A[x/y], \Delta_1) \cap \mathrm{L}(\Gamma_2, \Delta_2)$. As free variables are not considered to be part of the language, $L(\forall x A) = L(A[x/y])$.
2. $\langle(\Gamma_1; \Delta_1), (\Gamma_2, \forall x A; \Delta_2)\rangle$. This case can be argued analogously.

In the case of $\exists : r$, a similar argument goes through.

- The last rule application is an instance of $\forall : r$. Then it is of the form:

$$\frac{\Gamma \vdash \Delta, A[x/y]}{\Gamma \vdash \Delta, \forall x A} \forall : r$$

where y does not appear in Γ, Δ or A.

There are two possible partition schemes of $\Gamma \vdash \Delta, \forall x A$:

1. $\chi = \langle(\Gamma_1; \Delta_1, \forall x A), (\Gamma_2; \Delta_2)\rangle$. By the induction hypothesis, there exists an interpolant I of the partition $\langle(\Gamma_1; \Delta_1, A[x/y]), (\Gamma_2; \Delta_2)\rangle$ of the upper sequent. Hence for I, $\Gamma_1 \vdash \Delta_1, A[x/y], I$ and $I, \Gamma_2 \vdash \Delta_2$ are provable.
   As y does not occur in Γ or Δ and consequently by condition 3 does not occur in I, we may apply the $\forall : r$ rule to the former sequent to obtain $\Gamma_1 \vdash \Delta_1, \forall x A, I$. Hence I is an interpolant for χ as well.
2. $\langle(\Gamma_1; \Delta_1), (\Gamma_2; \Delta_2, \forall x A)\rangle$. This case can be argued analogously.

In the case of $\exists : l$, a similar argument goes through. □

This allows us to state the central theorem of this section:

Theorem 3.13. *Let Γ and Δ be sets of closed first-order formulas without equality and function symbols such that $\Gamma \cup \Delta$ is unsatisfiable. Then there is an interpolant for Γ and Δ.*

Proof. We show that there is an interpolant for $\Gamma \models \neg\Delta$, which by Proposition 1.4 proves the theorem. By the completeness of cut-free sequent calculus, there is a cut-free proof of $\Gamma \vdash \neg\Delta$. By Lemma 3.12, there is an interpolant I for the partition $\langle(\Gamma;), (; \neg\Delta)\rangle$. I is the desired interpolant for $\Gamma \models \neg\Delta$.

□

3.3 Proof by reduction

Using the results of the previous sections, we can now give a proof of the interpolation theorem:

Theorem 1.3 (Reverse Interpolation)**.** *Let* Γ *and* Δ *be sets of first-order formulas such that* $\Gamma \cup \Delta$ *is unsatisfiable. Then there exists a reverse interpolant for* Γ *and* Δ*.*

Proof. Since $\Gamma \cup \Delta$ is unsatisfiable, by Proposition 3.8, $\mathrm{T}_{\mathrm{Ax}}(\Gamma \cup \Delta)$ is unsatisfiable.

$$\begin{aligned}
\mathrm{T}_{\mathrm{Ax}}(\Gamma \cup \Delta) &\Leftrightarrow \{\mathrm{F}_{\mathrm{Ax}}(\mathrm{L}(\Gamma \cup \Delta)), \mathrm{E}_{\mathrm{Ax}}(\mathrm{L}(\Gamma \cup \Delta))\} \cup \mathrm{T}(\Gamma \cup \Delta) \\
&\Leftrightarrow \{\mathrm{F}_{\mathrm{Ax}}(\mathrm{L}(\Gamma) \cup \mathrm{L}(\Delta)), \mathrm{E}_{\mathrm{Ax}}(\mathrm{L}(\Gamma) \cup \mathrm{L}(\Delta))\} \cup \mathrm{T}(\Gamma) \cup \mathrm{T}(\Delta) \\
&\Leftrightarrow \{\mathrm{F}_{\mathrm{Ax}}(\mathrm{L}(\Gamma)) \wedge \mathrm{F}_{\mathrm{Ax}}(\mathrm{L}(\Delta)), \mathrm{E}_{\mathrm{Ax}}(\mathrm{L}(\Gamma)) \wedge \mathrm{E}_{\mathrm{Ax}}(\mathrm{L}(\Delta))\} \cup \mathrm{T}(\Gamma) \cup \mathrm{T}(\Delta) \\
&\Leftrightarrow \{\mathrm{F}_{\mathrm{Ax}}(\mathrm{L}(\Gamma)), \mathrm{E}_{\mathrm{Ax}}(\mathrm{L}(\Gamma))\} \cup \mathrm{T}(\Gamma) \cup \{\mathrm{F}_{\mathrm{Ax}}(\mathrm{L}(\Delta)), \mathrm{E}_{\mathrm{Ax}}(\mathrm{L}(\Delta))\} \cup \mathrm{T}(\Delta) \\
&\Leftrightarrow \mathrm{T}_{\mathrm{Ax}}(\Gamma) \cup \mathrm{T}_{\mathrm{Ax}}(\Delta)
\end{aligned}$$

Hence $\mathrm{T}_{\mathrm{Ax}}(\Gamma) \cup \mathrm{T}_{\mathrm{Ax}}(\Delta)$ is unsatisfiable as well. By Lemma 3.7.1 $\mathrm{T}_{\mathrm{Ax}}(\Gamma)$ and $\mathrm{T}_{\mathrm{Ax}}(\Delta)$ contain neither function symbols nor the equality symbol. Hence by Theorem 3.13, there is an interpolant I such that

1. $\mathrm{T}_{\mathrm{Ax}}(\Gamma) \models I$
2. $\mathrm{T}_{\mathrm{Ax}}(\Delta) \models \neg I$
3. $\mathrm{L}(I) \subseteq \mathrm{L}(\mathrm{T}_{\mathrm{Ax}}(\Gamma)) \cap \mathrm{L}(\mathrm{T}_{\mathrm{Ax}}(\Delta))$

We now show that $\mathrm{T}^{-1}(I)$ is an interpolant for Γ and Δ.

$\mathrm{T}_{\mathrm{Ax}}(\Gamma) \models I$ is equivalent to $\mathrm{T}_{\mathrm{Ax}}(\Gamma) \cup \{\neg I\}$ being unsatisfiable. Through the unfolding of $\mathrm{T}_{\mathrm{Ax}}(\Gamma)$, we get that $\{\mathrm{F}_{\mathrm{Ax}}(\mathrm{L}(\Gamma)), \mathrm{E}_{\mathrm{Ax}}(\mathrm{L}(\Gamma))\} \cup \mathrm{T}(\Gamma) \cup \{\neg I\}$ is unsatisfiable. This set of formulas can now be translated back to the original language with the equality symbol and function symbols. More formally, since $\mathrm{L}(\neg I) \subseteq \mathrm{L}(\mathrm{T}_{\mathrm{Ax}}(\Gamma))$, we can apply Proposition 3.8.2 by considering $\mathrm{T}(\Gamma) \cup \{\neg I\}$ as Φ to conclude that $\mathrm{T}^{-1}(\mathrm{T}(\Gamma) \cup \{\neg I\})$ is unsatisfiable. By pulling T^{-1} inward and an application of Lemma 3.5, we get that $\Gamma \cup \{\mathrm{T}^{-1}(\neg I)\} = \Gamma \cup \{\neg\, \mathrm{T}^{-1}(I)\}$ is unsatisfiable. Therefore $\Gamma \models \mathrm{T}^{-1}(I)$.

For Δ, an analogous argument goes through and so from $\mathrm{T}_{\mathrm{Ax}}(\Gamma) \models \neg I$ we can deduce that $\Delta \models \neg\, \mathrm{T}^{-1}(I)$.

By item 3, I is in the language $\mathrm{L}(\mathrm{T}_{\mathrm{Ax}}(\Gamma)) \cap \mathrm{L}(\mathrm{T}_{\mathrm{Ax}}(\Delta))$, which by Lemma 3.7.1 is $\mathrm{T}(\mathrm{L}(\Gamma)) \cap \mathrm{T}(\mathrm{L}(\Delta))$.

$$
\begin{aligned}
&\mathrm{T}(\mathrm{L}(\Gamma)) \cap \mathrm{T}(\mathrm{L}(\Delta)) = \\
&\quad \Big(\mathrm{L}(\Gamma) \ \cup\ \{E\} \ \cup\ \{F_f \mid f \in \mathrm{FS}(\Gamma)\}\Big) \setminus \Big(\{=\} \ \cup\ \mathrm{FS}(\Gamma)\Big) \ \cap \\
&\quad \Big(\mathrm{L}(\Delta) \ \cup\ \{E\} \ \cup\ \{F_f \mid f \in \mathrm{FS}(\Delta)\}\Big) \setminus \Big(\{=\} \ \cup\ \mathrm{FS}(\Delta)\Big) \\
&= \Big((\mathrm{L}(\Gamma) \cap \mathrm{L}(\Delta)) \ \cup\ \{E\} \ \cup\ \{F_f \mid f \in \mathrm{FS}(\Gamma) \cap \mathrm{FS}(\Delta)\}\Big) \setminus \Big(\{=\} \ \cup\ \mathrm{FS}(\Gamma) \ \cup\ \mathrm{FS}(\Delta)\Big) \\
&= \Big((\mathrm{L}(\Gamma) \cap \mathrm{L}(\Delta)) \ \cup\ \{E\} \ \cup\ \{F_f \mid f \in \mathrm{FS}(\mathrm{L}(\Gamma) \cap \mathrm{L}(\Delta))\}\Big) \setminus \Big(\{=\} \ \cup\ \mathrm{FS}(\mathrm{L}(\Gamma) \cap \mathrm{L}(\Delta))\Big) \\
&= \mathrm{T}(\mathrm{L}(\Gamma) \cap \mathrm{L}(\Delta))
\end{aligned}
$$

As I is in the language $\mathrm{T}(\mathrm{L}(\Gamma) \cap \mathrm{L}(\Delta))$, by Lemma 3.7.2, $\mathrm{T}^{-1}(I)$ is in the language $\mathrm{L}(\Gamma) \cap \mathrm{L}(\Delta)$. □

Chapter 4

Interpolant extraction from resolution proofs in two phases

In [Hua95], Huang proposes an algorithm for computing interpolants of two sets of first-order formulas Γ and Δ, where $\Gamma \cup \Delta$ is unsatisfiable, by traversing a resolution refutation of $\Gamma \cup \Delta$. We present his proof in a modified form. The central difference lies in the treatment of the interplay of substitutions and liftings. While in [Hua95], propositional deductions are employed where only trivial substitutions occur, we provide a method which allows for commuting substitutions and liftings under certain conditions.

See also TODO: for a comments on the original proof.

4.1 Layout of the proof

The underlying algorithm produces in the first phase propositional interpolants inductively for every clause which occurs in the resolution refutation. These interpolants are propositional in the sense that they only obey the language restriction on predicates and may contain colored terms. The propositional interpolant assigned to the last clause, the empty clause, is a propositional interpolant for the initial clause sets.

The second phase of the algorithm addresses the colored terms still contained in the propositional interpolant. These are eliminated (lifted) by replacing them with bound variables whose quantifiers are subject to a certain ordering.

4.2 Extraction of propositional interpolants

We define a procedure PI, which produces propositional interpolants from resolution refutations. It only differs marginally from the "Interpolation Algorithm" in [Hua95]. TODO: adapt this when final version is decided

Definition 4.1 (Propositional interpolant extraction.)**.** Let π be a resolution refutation of $\Gamma \cup \Delta$. PI(π) is defined to be PI($\square$), where $\square$ is the empty clause derived in π.

For a clause C in π, $\mathrm{PI}(C)$ is defined as follows:

Base case. If $C \in \Gamma$, $\mathrm{PI}(C) \stackrel{\text{def}}{=} \bot$. If otherwise $C \in \Delta$, $\mathrm{PI}(C) \stackrel{\text{def}}{=} \top$.

Resolution. If the clause C is the result of a resolution step of $C_1 : D \vee l$ and $C_2 : E \vee \neg l'$ using a unifier σ such that $l\sigma = l'\sigma$, then $\mathrm{PI}(C)$ is defined as follows:

1. If l is Γ-colored: $\mathrm{PI}(C) \stackrel{\text{def}}{=} [\mathrm{PI}(C_1) \vee \mathrm{PI}(C_2)]\sigma$
2. If l is Δ-colored: $\mathrm{PI}(C) \stackrel{\text{def}}{=} [\mathrm{PI}(C_1) \wedge \mathrm{PI}(C_2)]\sigma$
3. If l is grey: $\mathrm{PI}(C) \stackrel{\text{def}}{=} [(l \wedge \mathrm{PI}(C_2)) \vee (\neg l' \wedge \mathrm{PI}(C_1))]\sigma$

Factorisation. If the clause C is the result of a factorisation of $C_1 : l \vee l' \vee D$ using a unifier σ such that $l\sigma = l'\sigma$, then $\mathrm{PI}(C) = \mathrm{PI}(C_1)\sigma$.

Paramodulation. Suppose the clause C is the result of a paramodulation of $C_1 : s = t \vee C$ and $C_2 : D[r]$ using a unifier σ such that $r\sigma = s\sigma$. Let $h[r]$ be the maximal colored term in which r occurs in $D[r]$. Then $\mathrm{PI}(C)$ is defined according to the following case distinction:

1. If $h[r]$ is Δ-colored and $h[r]$ occurs more than once in $D[r] \vee \mathrm{PI}(D[r])$:
   $\mathrm{PI}(C) \stackrel{\text{def}}{=} [(s = t \wedge \mathrm{PI}(C_2)) \vee (s \neq t \wedge \mathrm{PI}(C_1))]\sigma \vee (s = t \wedge h[s] \neq h[t])\sigma$
2. If $h[r]$ is Γ-colored and $h[r]$ occurs more than once in $D[r] \vee \mathrm{PI}(D[r])$:
   $\mathrm{PI}(C) \stackrel{\text{def}}{=} [(s = t \wedge \mathrm{PI}(C_2)) \vee (s \neq t \wedge \mathrm{PI}(C_1))]\sigma \wedge (s \neq t \vee h[s] = h[t])\sigma$
3. If r does not occur in a colored term in $D[r]$ or $s \equiv t$:
   $\mathrm{PI}(C) \stackrel{\text{def}}{=} [(s = t \wedge \mathrm{PI}(C_2)) \vee (s \neq t \wedge \mathrm{PI}(C_1))]\sigma$ △

The difference between PI and the "Interpolation Algorithm" in [Hua95] lies in the definition for the case of paramodulation. TODO:

4.3 Lifting of colored symbols

As PI only fixes the propositional structure of the interpolant but still contains colored symbol, we define a procedure which replaces colored terms by lifting variables.

Definition 4.2 (Lifting)**.** Let Γ and Δ be sets of first-order formulas, φ a formula or a term, $Z = \{\zeta_1, \ldots, \zeta_n\}$ the maximal Φ-terms for $\Phi \in \{\Gamma, \Delta\}$ in φ and $z_{\zeta_1}, \ldots, z_{\zeta_n}$ fresh variables, referred to as Φ-*lifting variables* or *lifting variables* if the coloring is clear from the context.

Then $\ell^z_\Phi[\varphi]$ denotes $\ell^z_\Phi[\varphi, Z]$ which is defined as follows:

$$\ell^z_\Phi[\varphi, Z] \stackrel{\text{def}}{=} \begin{cases} \varphi & Z = \varnothing \\ \ell^z_\Phi[\varphi\{\zeta_i / z_{\zeta_i}\}, Z \backslash \{\zeta_i\}] & \zeta_i \in Z \text{ such that } \zeta_i \text{ is not subterm of another term in } Z \end{cases}$$

To simplify the syntax, we sometimes write $\ell_\Phi[\varphi]$ or $\ell[\varphi]$ if the lifting variables or the lifting variables and the color of the terms to lift respectively is clear from the context or not of the essence. △

We usually lift Δ-terms by variables with the letter x and Γ-terms with the letter y. If the lifting is not specific to a color, we use variables with the letter z.

Some elementary properties of liftings are described by the following lemmas:

Lemma 4.3 (Commutativity of lifting and logical operators)**.** *Let A and B be first-order formulas and s and t be terms. Then it holds that:*

1. $\ell^z_\Phi[\neg A] \Leftrightarrow \neg\ell^z_\Phi[A]$
2. $\ell^z_\Phi[A \circ B] \Leftrightarrow (\ell^z_\Phi[A] \circ \ell^z_\Phi[B])$ *for* $\circ \in \{\wedge, \vee\}$
3. $\ell^z_\Phi[s = t] \Leftrightarrow (\ell^z_\Phi[s] = \ell^z_\Phi[t])$

For the proof, we also require a means of commuting substitutions and liftings. This however can not be achieved in a direct manner. The following examples illustrate that in general for a term t, it is not the case that $\ell^z_\Phi[t\sigma] = \ell^z_\Phi[t]\sigma$. However Lemma 4.5 defines a substitution σ' such that $\ell^z_\Phi[t\sigma] = \ell^z_\Phi[t]\sigma'$.

Example 4.4.A. Let $t = f(x)$ be a Φ-term and $\sigma = \{x \mapsto a\}$. Then $\ell^z_\Phi[t\sigma] = \ell^z_\Phi[f(x)\sigma] = \ell^z_\Phi[f(a)] = z_{f(a)}$. However $\ell^z_\Phi[t]\sigma = \ell^z_\Phi[f(x)]\sigma = z_{f(x)}\sigma = z_{f(x)}$. △

Example 4.4.B. Let $t = x$ be a variable and $\sigma = \{x \mapsto c\}$, where c is a Φ-term. Then $\ell^z_\Phi[t\sigma] = \ell^z_\Phi[x\sigma] = \ell^z_\Phi[c] = z_c$. But $\ell^z_\Phi[t]\sigma = \ell^z_\Phi[x]\sigma = x\sigma = c$. △

Lemma 4.5 (Commutativity of lifting and substitution)**.** *Let C be a clause and σ a substitution such that no lifting variable occurs in C or σ. Define σ' with* $\mathrm{dom}(\sigma') = \mathrm{dom}(\sigma) \cup \{z_t \mid t\sigma \neq t\}$ *such that for a variable z,*

$$x\sigma' = \begin{cases} z_{t\sigma} & \text{if } x = z_t \text{ and } t\sigma \neq t \\ \ell^z_\Phi[x\sigma] & \text{otherwise} \end{cases}$$

Then $\ell^z_\Phi[C\sigma] = \ell^z_\Phi[C]\sigma'$.

Proof. As substitutions and liftings only affect the terms of a clause, it suffices to show that $\ell^z_\Phi[t\sigma] = \ell^z_\Phi[t]\sigma'$ for for a term t in C. More precisely, only variables of $\mathrm{dom}(\sigma)$ and maximal Δ-terms are affected. We show that for terms t of either kind that $\ell^z_\Phi[t\sigma] = \ell^z_\Phi[t]\sigma'$ holds, which proves the lemma.

Let y be a variable in $\mathrm{dom}(\sigma)$, which occurs in C. Then $\ell^z_\Phi[y]\sigma' = y\sigma' = \ell^z_\Phi[y\sigma]$.

Let t be a maximal Φ-term in C. Then $\ell^z_\Phi[t\sigma] = z_{t\sigma}$. We show that $z_{t\sigma} = \ell^z_\Phi[t]\sigma'$.

Suppose that $t\sigma = t$. Then $\ell^z_\Phi[t]\sigma' = z_t\sigma' = z_t = z_{t\sigma}$. Note that z_r must not occur in t for some term r, as $z_r\sigma = z_r$, but potentially $z_r\sigma' \neq z_r$.

Otherwise it is the case that $t\sigma \neq t$. Then $\ell^z_\Phi[t]\sigma' = z_t\sigma'$, and by the definition of σ', $z_t\sigma' = z_{t\sigma}$. □

Example 4.6. Let M be a model and c and d be Φ-colored constants such that $M \models c = d$. Then

?????

△

Lemma 4.7. *Let M be a model, E a formula and s and t terms such that $M \models \ell^x_\Delta[s] = \ell^x_\Delta[t]$. Let $h[t]$ be a maximal Δ-colored term containing t at p in $E[t]_p$, if such a term exists. Then it holds that:*

- *If $h[t]$ does not exists, then $M \models \ell^x_\Delta[E[s]_p] \Leftrightarrow M \models \ell^x_\Delta[E[s]_p]$.*
- *Otherwise $M \models \ell^x_\Delta[E[s]_p] \Leftrightarrow M \models \ell^x_\Delta[E[s]_p]$ or $M \models \ell^x_\Delta[h[s]] \neq \ell^x_\Delta[h[t]]$ holds.*

Proof. Suppose that the position p in $E[s]_p$ is not contained in a Δ-colored term. Then $\ell^x_\Delta[E[t]_p]$ and $\ell^x_\Delta[E[s]_p]$ only differ at position p, where for the first, $\ell^x_\Delta[t]$ is at p, and for the latter, $\ell^x_\Delta[s]$ is at p. But in M, they are interpreted the same way, hence $M \models \ell^x_\Delta[E[s]_p] \Leftrightarrow M \models \ell^x_\Delta[E[t]_p]$.

Otherwise the position p in $E[t]_p$ is contained in the maximal Δ-colored term $h[t]$. Suppose that $M \models \ell^x_\Delta[h[s]] = \ell^x_\Delta[h[t]]$ as otherwise we would be done. But then $M \models \ell^x_\Delta[E[s]_p] \Leftrightarrow M \models \ell^x_\Delta[E[t]_p]$. □

4.4 Main lemma

By lifting the propositional interpolant, we are able to already obtain a formula partially fulfilling the requirements for interpolants:

Lemma 4.8. *Let π be a resolution refutation of $\Gamma \cup \Delta$. Then for a clause C in π, $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C) \vee C]$.*

Before we proceed with the proof of this lemma, we give a corollary which demonstrates that PI extracts propositional interpolants in the sense that besides potentially containing colored terms, they are proper interpolants:

Corollary 4.9. *Let π be a resolution refutation of $\Gamma \cup \Delta$. Then*

1. $\Gamma \models \mathrm{PI}(\pi)$
2. $\Delta \models \mathrm{PI}(\pi)$
3. $\mathrm{PS}(\mathrm{PI}(\pi)) \subseteq \mathrm{PS}(\Gamma) \cap \mathrm{PS}(\Delta)$.

Proof. By the definition of PI, $\mathrm{PI}(\pi)$ denotes $\mathrm{PI}(\square)$, where $\square$ is the empty clause derived in PI. By Lemma 4.8, we get that $\Gamma \models \ell^x_\Delta[\mathrm{PI}(\pi)]$. As the lifting replaces terms by variables which are then implicitly universally quantified, $\mathrm{PI}(\pi)$ is an instance of $\ell^x_\Delta[\mathrm{PI}(\pi)]$. Therefore $\Gamma \models \mathrm{PI}(\pi)$.

Item 2 can be argued analogously as π is also a refutation of $\hat{\Gamma} \cup \hat{\Delta}$, where $\hat{\Gamma} = \Delta$ and $\hat{\Delta} = \Gamma$.

By the construction of PI, only grey predicates are added. □

Proof of Lemma 4.8. We proceed by induction on the resolution refutation of the strengthening $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C) \vee C_\Gamma]$, where D_Φ denotes the clause created from D by removing all literals which are not contained $\mathrm{L}(\Phi)$.

Base case. Either $C \in \Gamma$, then $\ell^x_\Delta[C] = C$ and $\Gamma \models C$. Otherwise $C \in \Delta$ and $\mathrm{PI}(C) = \top$.

Resolution. Suppose the last rule application is an instance of resolution. Then it is of the following form:

$$\frac{C_1 : D \vee l \qquad C_2 : E \vee \neg l'}{C : (D \vee E)\sigma} \quad l\sigma = l'\sigma$$

By the induction hypothesis, we can assume that $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1) \vee (D \vee l)_\Gamma]$ and $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_2) \vee (E \vee \neg l')_\Gamma]$, which by Lemma 4.3 implies $(\circ)$ $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1)] \vee \ell^x_\Delta[D_\Gamma] \vee \ell^x_\Delta[l_\Gamma]$ and $(*)$ $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_2)] \vee \ell^x_\Delta[E_\Gamma] \vee \neg\ell^x_\Delta[l'_\Gamma]$. Let σ' be defined as in Lemma 4.5.

We proceed by a case distinction on the color of l:

1. l is Γ-colored. Then $\mathrm{PI}(C) = [\mathrm{PI}(C_1) \vee \mathrm{PI}(C_2)]\sigma$.
   Since $\sigma = \mathrm{mgu}(l, l')$, $l\sigma \equiv l'\sigma$ and therefore $\ell^x_\Delta[l\sigma] = \ell^x_\Delta[l'\sigma]$. As by Lemma 4.5 $\ell^x_\Delta[l\sigma] = \ell^x_\Delta[l]\sigma'$ and $\ell^x_\Delta[l'\sigma] = \ell^x_\Delta[l']\sigma'$, we get $\ell^x_\Delta[l]\sigma' = \ell^x_\Delta[l']\sigma'$. Hence by applying σ' to $(\circ)$ and $(*)$ (note that $l_\Gamma = l$ and $l'_\Gamma = l'$ as they are Γ-colored), we can perform a resolution step on $\ell^x_\Delta[l]\sigma'$ and obtain $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1)]\sigma' \vee \ell^x_\Delta[D_\Gamma]\sigma' \vee \ell^x_\Delta[\mathrm{PI}(C_2)]\sigma' \vee \ell^x_\Delta[E_\Gamma]\sigma'$. Now we apply Lemma 4.5 and Lemma 4.3 in the other direction and get that $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1)\sigma \vee \mathrm{PI}(C_2)\sigma \vee (D \vee E)_\Gamma\sigma]$. This however is nothing else than $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C) \vee C_\Gamma]$.
2. l is Δ-colored. Then $\mathrm{PI}(C) = [\mathrm{PI}(C_1) \wedge \mathrm{PI}(C_2)]\sigma$.
   As l and l' are Δ-colored, we can simplify $(\circ)$ and $(*)$ as follows and apply σ': $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1)]\sigma' \vee \ell^x_\Delta[D_\Gamma]\sigma'$ and $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_2)]\sigma' \vee \ell^x_\Delta[E_\Gamma]\sigma'$. These together imply that $\Gamma \models \Big(\ell^x_\Delta[\mathrm{PI}(C_1)]\sigma' \wedge \ell^x_\Delta[\mathrm{PI}(C_2)]\sigma'\Big) \vee \ell^x_\Delta[D_\Gamma]\sigma' \vee \ell^x_\Delta[E_\Gamma]\sigma'$. By Lemma 4.3 and Lemma 4.5, this is equivalent to $\Gamma \models \ell^x_\Delta[(\mathrm{PI}(C_1) \wedge \mathrm{PI}(C_2))\sigma \vee (D \vee E)_\Gamma\sigma]$, which is nothing else than $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C) \vee C_\Gamma]$.
3. l is grey. Then $\mathrm{PI}(C) = [(l \wedge \mathrm{PI}(C_2)) \vee (\neg l' \wedge \mathrm{PI}(C_1))]\sigma$.
   We show that $\Gamma \models \ell^x_\Delta[\Big((l \wedge \mathrm{PI}(C_2)) \vee (\neg l' \wedge \mathrm{PI}(C_2)) \vee D_\Gamma \vee E_\Gamma\Big)\sigma]$, for which by Lemma 4.3 and Lemma 4.5 it suffices to show that $\Gamma \models \big(\ell^x_\Delta[l]\sigma' \wedge \ell^x_\Delta[\mathrm{PI}(C_2)]\sigma'\big) \vee \big(\neg\ell^x_\Delta[l']\sigma' \wedge \ell^x_\Delta[\mathrm{PI}(C_2)]\sigma'\big) \vee \ell^x_\Delta[D_\Gamma]\sigma' \vee \ell^x_\Delta[E_\Gamma]\sigma'$.
   Suppose for a model M of Γ that $M \models \ell^x_\Delta[D_\Gamma]\sigma'$ and $M \models \ell^x_\Delta[E_\Gamma]\sigma'$ as otherwise we are done. But then by $(\circ)$ and $(*)$, we get that $M \models \ell^x_\Delta[\mathrm{PI}(C_1)]\sigma' \vee \ell^x_\Delta[l]\sigma'$ and $M \models \ell^x_\Delta[\mathrm{PI}(C_2)]\sigma' \vee \neg\ell^x_\Delta[l']\sigma'$. As observed in case 1, $\ell^x_\Delta[l]\sigma' = \ell^x_\Delta[l']\sigma'$. We obtain the result by a case distinction on the truth value of $\ell^x_\Delta[l]\sigma'$.

Factorisation. Suppose the last rule application is an instance of factorisation. Then it is of the following form:

$$\frac{C_1 : l \vee l' \vee D}{C : (l \vee D)\sigma} \quad \sigma = \mathrm{mgu}(l, l')$$

The induction hypothesis gives that $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1) \vee l \vee l' \vee D]$. Let σ' be defined as in Lemma 4.5. Then $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1) \vee l \vee l' \vee D]\sigma'$ and by Lemma 4.5 and Lemma 4.3, $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1)\sigma] \vee \ell^x_\Delta[l\sigma] \vee \ell^x_\Delta[l'\sigma] \vee \ell^x_\Delta[D\sigma]$.

As $\sigma = \mathrm{mgu}(l, l')$, $l\sigma \equiv l'\sigma$ and hence $\ell^x_\Delta[l\sigma] = \ell^x_\Delta[l'\sigma]$. But then we can apply a factorisation step and get that $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1)\sigma] \vee \ell^x_\Delta[l\sigma] \vee \ell^x_\Delta[D\sigma]$ which by Lemma 4.5 and Lemma 4.3 is equivalent to $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1)\sigma \vee l\sigma \vee D\sigma]$. This in turn is nothing else than $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C) \vee C]$.

Paramodulation. Suppose the last rule application is an instance of paramodulation. Then it is of the following form:

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

By the induction hypothesis, we get that $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1) \vee (D \vee s = t)_\Gamma]$ and $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_2) \vee (E[r]_p)_\Gamma]$. This is by Lemma 4.3 equivalent to ($\circ$) $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1)] \vee \ell^x_\Delta[D_\Gamma] \vee \ell^x_\Delta[s] = \ell^x_\Delta[t]$ and ($*$) $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_2)] \vee \ell^x_\Delta[(E[r]_p)_\Gamma]$ respectively.

We distinguish two cases:

1. Suppose s does not occur in a maximal Δ-term $h[s]$ in $E[s]_p$, which occurs more than once in $\mathrm{PI}(C_2) \vee (E[r]_p)_\Gamma$.

   Let M be a model of Γ. First, assume that $M \models \ell^x_\Delta[s] \neq \ell^x_\Delta[t]$. Then by ($\circ$) we have that $M \models \ell^x_\Delta[\mathrm{PI}(C_1)] \vee \ell^x_\Delta[D_\Gamma]$. By applying σ' and Lemma 4.5, we hence can conclude from $M \models \ell^x_\Delta[s\sigma] \neq \ell^x_\Delta[t\sigma]$ that $M \models \ell^x_\Delta[\mathrm{PI}(C_1)\sigma] \vee \ell^x_\Delta[D_\Gamma\sigma]$.

   Second, assume to the contrary that $M \models \ell^x_\Delta[s] = \ell^x_\Delta[t]$. We distinguish cases:

   - Suppose that s does not occur in a maximal Δ-term in $(E[s]_p)_\Gamma$. Then by Lemma 4.7 $M \models \ell^x_\Delta[(E[t]_p)_\Gamma] \Leftrightarrow M \models \ell^x_\Delta[(E[s]_t)_\Gamma]$.
     Due to $\sigma = \mathrm{mgu}(s, r)$, $s\sigma \equiv r\sigma$. Suppose they are both not Δ-colored. Then the lifting does not affect them and $\ell^x_\Delta[(E[s]_p)_\Gamma\sigma] \equiv \ell^x_\Delta[(E[r]_p)_\Gamma\sigma]$. Otherwise the lifting will replace them with the same variable and we as well get that $\ell^x_\Delta[(E[s]_p)_\Gamma\sigma] \equiv \ell^x_\Delta[(E[r]_p)_\Gamma\sigma]$. Hence $M \models \ell^x_\Delta[(E[t]_p)_\Gamma\sigma] \Leftrightarrow M \models \ell^x_\Delta[(E[r]_p)_\Gamma\sigma]$
   - Otherwise r occurs in a maximal Δ-term $h[r]$ in $(E[r]_p)_\Gamma$, but $h(r)$ does not occur elsewhere in $\mathrm{PI}(C_2) \vee (E[r]_p)_\Gamma$. Then the lifting variable x_r occurs only once in ($*$). Hence Γ does not pose any restriction on x_r and we can substitute it in by x_t. So in M we have that $M \models \ell^x_\Delta[(E[t]_p)_\Gamma] \Leftrightarrow M \models \ell^x_\Delta[(E[r]_p)_\Gamma]$.

Hence in any of the cases, by $(*)$ and by applying σ' and Lemma 4.5 it follows from $M \models \ell^x_\Delta[s\sigma] = \ell^x_\Delta[t\sigma]$ that $M \models \ell^x_\Delta[\mathrm{PI}(C_2)\sigma] \vee \ell^x_\Delta[(E[t]_p)_\Gamma\sigma]$.

In conclusion, we can derive that $\Gamma \models (\ell^x_\Delta[s\sigma] \neq \ell^x_\Delta[t\sigma] \wedge (\ell^x_\Delta[\mathrm{PI}(C_1)\sigma] \vee \ell^x_\Delta[D_\Gamma]\sigma)) \vee (\ell^x_\Delta[s\sigma] = \ell^x_\Delta[t\sigma] \wedge (\ell^x_\Delta[\mathrm{PI}(C_2)\sigma] \vee \ell^x_\Delta[(E[t]_p)_\Gamma\sigma]))$. This however implies that $\Gamma \models \Big(\ell^x_\Delta[s\sigma] \neq \ell^x_\Delta[t\sigma] \wedge (\ell^x_\Delta[\mathrm{PI}(C_1)\sigma])\Big) \vee \Big(\ell^x_\Delta[s\sigma] = \ell^x_\Delta[t\sigma] \wedge (\ell^x_\Delta[\mathrm{PI}(C_2)\sigma])\Big) \vee \Big(\ell^x_\Delta[D_\Gamma\sigma] \vee \ell^x_\Delta[(E[t]_p)_\Gamma\sigma]\Big)$, which by Lemma 4.3 is nothing else than $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C) \vee C]$.

2. Otherwise s occurs in a maximal Δ-term $h[s]_q$ in $E[s]_p$, which occurs more than once in $E[s]_p$.

   Then a similar line of argument as in case 1 can be employed, with the difference that the application of Lemma 4.7 in the case of $M \models \ell^x_\Delta[s] = \ell^x_\Delta[t]$ yields the additional possibility that $M \models \ell^x_\Delta[h[s]] \neq \ell^x_\Delta[h[t]]$. Hence we arrive at:
   $\Gamma \models \Big(\ell^x_\Delta[s]\sigma' = \ell^x_\Delta[t]\sigma' \wedge \ell^x_\Delta[\mathrm{PI}(C_2)]\sigma'\Big) \vee \Big(\ell^x_\Delta[s]\sigma' \neq \ell^x_\Delta[t]\sigma' \wedge \ell^x_\Delta[\mathrm{PI}(C_1)]\sigma'\Big) \vee \Big(\ell^x_\Delta[s]\sigma' = \ell^x_\Delta[t]\sigma' \wedge (\ell^x_\Delta[h[s]]\sigma') \neq (\ell^x_\Delta[h[t]]\sigma')\Big) \vee \Big(\ell^x_\Delta[D_\Gamma]\sigma' \vee \ell^x_\Delta[(E[t]_p)_\Gamma]\sigma'\Big)$
   This however is by Lemma 4.5 and Lemma 4.3 equivalent to $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C) \vee C]$.

□

4.5 Quantifying over lifting variables

The interpolant extraction procedure PI exhibits a convenient property which is termed *symmetry* in [DKPW10, Definition 5] and will be used to show that Lemma 4.8 can easily be generalised to a result for Δ, which is formulated in the Corollary 4.11 below. The symmetry-property of PI can be stated formally as follows:

Lemma 4.10. *Let π be a resolution refutation of $\Gamma \cup \Delta$ and $\hat{\pi}$ be π with $\hat{\Gamma} = \Delta$ and $\hat{\Delta} = \Gamma$. Then* $\mathrm{PI}(\pi) \Leftrightarrow \neg\,\mathrm{PI}(\hat{\pi})$.

Proof. We prove this lemma by induction on π. Let $\hat{\varphi}$ denote the clause/formula/literal/term in $\hat{\pi}$ corresponding to the clause/formula/literal/term φ in π.

Base case. If $C \in \Gamma$, then $C' \in \Delta'$ and $\mathrm{PI}(C) = \bot \Leftrightarrow \neg\top = \neg\,\mathrm{PI}(C')$. The case for $C \in \Delta$ is analogous.

Resolution. If the clause C is the result of a resolution step of $C_1 : D \vee l$ and $C_2 : E \vee \neg l'$ using a unifier σ such that $l\sigma = l'\sigma$, then by induction hypothesis, we get that $\mathrm{PI}(C_i) = \neg\,\mathrm{PI}(C_i')$ for $i \in \{1, 2\}$.

We distinguish the following cases:

1. l is Γ-colored. Then $\hat{l}$ is Δ-colored.

$$\begin{aligned}\mathrm{PI}(C) &= \mathrm{PI}(C_1) \vee \mathrm{PI}(C_2)\\ &\Leftrightarrow \neg(\neg\,\mathrm{PI}(C_1) \wedge \neg\,\mathrm{PI}(C_2))\\ &= \neg(\mathrm{PI}(\hat{C}_1) \wedge \mathrm{PI}(\hat{C}_2))\\ &= \neg\,\mathrm{PI}(\hat{C})\end{aligned}$$

2. l is Δ-colored. This case can be argued analogously.
3. l is grey. Then $\hat{l}$ is grey.

$$\begin{aligned}\mathrm{PI}(C) &= [(l \wedge \mathrm{PI}(C_2)) \vee (\neg l' \wedge \mathrm{PI}(C_1))]\sigma\\ &= (l\sigma \wedge \mathrm{PI}(C_2)\sigma) \vee (\neg l'\sigma \wedge \mathrm{PI}(C_1)\sigma)\\ &\Leftrightarrow (\neg l\sigma \vee \mathrm{PI}(C_2)\sigma) \wedge (l'\sigma \vee \mathrm{PI}(C_1)\sigma)\\ &\Leftrightarrow \neg[(l\sigma \wedge \neg\,\mathrm{PI}(C_2)\sigma) \vee (\neg l'\sigma \wedge \neg\,\mathrm{PI}(C_1)\sigma)]\\ &= \neg[(\hat{l}\sigma \wedge \neg\,\mathrm{PI}(C_2)\sigma) \vee (\neg \hat{l}'\sigma \wedge \neg\,\mathrm{PI}(C_1)\sigma)\\ &= \neg[(\hat{l} \wedge \neg\,\mathrm{PI}(C_2)) \vee (\neg \hat{l}' \wedge \neg\,\mathrm{PI}(C_1))]\sigma\\ &= \neg[(\hat{l} \wedge \mathrm{PI}(\hat{C}_2)) \vee (\neg \hat{l}' \wedge \mathrm{PI}(\hat{C}_1))]\sigma\\ &= \neg\,\mathrm{PI}(\hat{C})\end{aligned}$$

Factorisation. Suppose the clause C is the result of a factorisation of $C_1 : l \vee l' \vee D$. Then $\mathrm{PI}(C) = \mathrm{PI}(C_1)\sigma$ and the induction hypothesis gives the result.

Paramodulation. Suppose the clause C is the result of a paramodulation of $C_1 : s = t \vee C$ and $C_2 : D[r]$ using a unifier σ such that $r\sigma = s\sigma$. We distinguish the following cases:

1. r occurs in a maximal Δ-term $h[r]$ in $D[r]$ and $h[r]$ occurs more than once in $D[r] \vee \mathrm{PI}(D[r])$. Then $\hat{r}$ occurs in a maximal Γ-term $\hat{h}[r]$ in $\hat{D}[r]$ and $\hat{h}[r]$ occurs more than once in $\hat{D}[r] \vee \mathrm{PI}(\hat{D}[r])$.

$$\begin{aligned}\mathrm{PI}(C) &= [(s = t \wedge \mathrm{PI}(C_2)) \vee (s \neq t \wedge \mathrm{PI}(C_1))]\sigma \vee (s = t \wedge h[s] \neq h[t])\sigma\\ &= [(s = t \wedge \neg\,\mathrm{PI}(\hat{C}_2)) \vee (s \neq t \wedge \neg\,\mathrm{PI}(\hat{C}_1))]\sigma \vee (s = t \wedge h[s] \neq h[t])\sigma\\ &\Leftrightarrow \neg[(s \neq t \vee \mathrm{PI}(\hat{C}_2)) \wedge (s = t \vee \mathrm{PI}(\hat{C}_1))]\sigma \wedge \neg(s \neq t \vee h[s] = h[t])\sigma\\ &\Leftrightarrow \neg[(s = t \wedge \mathrm{PI}(\hat{C}_2)) \vee (s \neq t \wedge \mathrm{PI}(\hat{C}_1))]\sigma \wedge \neg(s \neq t \vee h[s] = h[t])\sigma\\ &= \neg\,\mathrm{PI}(\hat{C})\end{aligned}$$

2. r occurs in a maximal Γ-term $h(r)$ in $D[r]$ and $h(r)$ occurs more than once in $D[r] \vee \mathrm{PI}(D[r])$. This case can be argued analogously.

3. Otherwise:

$$\begin{aligned}\mathrm{PI}(C) &= [(s = t \wedge \mathrm{PI}(C_2)) \vee (s \neq t \wedge \mathrm{PI}(C_1))]\sigma \\ &= [(s = t \wedge \neg\,\mathrm{PI}(\hat{C}_2)) \vee (s \neq t \wedge \neg\,\mathrm{PI}(\hat{C}_1))]\sigma \\ &\Leftrightarrow \neg[(s \neq t \vee \mathrm{PI}(\hat{C}_2)) \wedge (s = t \vee \mathrm{PI}(\hat{C}_1))]\sigma \\ &\Leftrightarrow \neg[(s = t \wedge \mathrm{PI}(\hat{C}_2)) \vee (s \neq t \wedge \mathrm{PI}(\hat{C}_1))]\sigma \\ &= \neg\,\mathrm{PI}(\hat{C})\end{aligned}$$ □

Corollary 4.11. *Let π be a resolution refutation of $\Gamma \cup \Delta$. Then $\Delta \models \ell^x_\Gamma[\neg\,\mathrm{PI}(C) \vee C]$ for C in π.*

Proof. Build $\hat{\pi}$ from π using $\hat{\Gamma} = \Delta$ and $\hat{\Delta} = \Gamma$ as initial clause sets. By Lemma 4.8, $\hat{\Gamma} \models \ell^x_{\hat{\Delta}}[\mathrm{PI}(\hat{C}) \vee \hat{C}]$ for $\hat{C}$ in $\hat{\pi}$. By Lemma 4.10, $\hat{\Gamma} \models \ell^x_{\hat{\Delta}}[\neg\,\mathrm{PI}(C) \vee \hat{C}]$ for the clause C in π corresponding to $\hat{C}$ in $\hat{\pi}$. This however is nothing else than $\Delta \models \ell^x_\Gamma[\neg\,\mathrm{PI}(C) \vee C]$. □

Lemma 4.12. *$\ell^y_\Gamma[\ell^x_\Delta[C]]$ and $\ell^{x'}_\Delta[\ell^{y'}_\Gamma[C]]$ differ only in the naming of the variables replacing maximal colored terms.*

Proof. Suppose a term t in C is affected by a lifting. We only need to consider maximal colored terms as grey terms are not affected by the liftings. Without loss of generality let t be a maximal Δ-colored term.

Let Φ be the positions of maximal occurrences of t. Then in the left hand side, exactly all terms at positions Φ are replaced by x_i for some i.

In the right hand side, all terms at positions Φ are replaced by $\ell^{y'}_\Gamma[t]$ first. However after this step, all these terms are equal to $\ell^{y'}_\Gamma[t]$, and as all distinct maximal Γ-terms are replaced by distinct variables, no other maximal colored term is equal to $\ell^{y'}_\Gamma[t]$. Hence exactly the terms at positions Φ are replaced by the same variable x'_j for some j. □

Using the already given results, it only remains to quantify over the lifting variables of the lifted propositional interpolant appropriately to arrive at a proper interpolant:

Theorem 4.13. *Let π be a resolution refutation of $\Gamma \cup \Delta$ and $z_1, \ldots, z_n$ be the variables which replace the colored terms in $\ell^y_\Gamma[\ell^x_\Delta[\mathrm{PI}(\pi)]]$ ordered by their length. Then $Q_1 z_1 \ldots Q_n z_n\, \ell^y_\Gamma[\ell^x_\Delta[\mathrm{PI}(\pi)]]$, where Q_i is $\forall$ ($\exists$) if z_i replaces a Δ (Γ)-term, is an interpolant.*

Proof. By Lemma 4.8, $\Gamma \models \forall x_1 \ldots \forall x_m\, \ell^x_\Delta[\mathrm{PI}(\pi)]$ where m is the number of maximal Δ-colored terms in $\mathrm{PI}(\pi)$.

A term in $\ell^x_\Delta[\mathrm{PI}(\pi)]$ is either x_i, $1 \leq i \leq m$, a grey term or a Γ-terms. Let t be a maximal Γ-term in $\ell^x_\Delta[\mathrm{PI}(\pi)]$ and $x_{j_1}, \ldots, x_{j_k}$ the variables replacing Δ-terms in t. Note that the Δ-terms, which are replaced by $x_{j_1}, \ldots, x_{i_{j_k}}$ respectively, are each of strictly smaller size than t as they are strict subterms of t.

In $\ell^y_\Gamma[\ell^x_\Delta[\mathrm{PI}(\pi)]]$, t is replaced by some z_j, which is existentially quantified. Hence t is a witness for z_j as due to the quantifier ordering, the existential quantification of z_j is in the scope of the quantifiers of $x_{j_1}, \ldots, x_{j_k}$ respectively. Therefore $\Gamma \models Q_1 z_1 \ldots Q_n z_n\, \ell^y_\Gamma[\ell^x_\Delta[\mathrm{PI}(\pi)]]$.

By Corollary 4.11 $\Delta \models \forall y_1 \ldots \forall y_k\, \neg \ell^y_\Gamma[\mathrm{PI}(\pi)]$, where k is the number of Γ-colored terms in $\mathrm{PI}(\pi)$. By a similar line of argumentation as above, we can replace the maximal Δ-terms by existentially quantified variables and arrive at $\Delta \models \overline{Q}_1 z_1 \ldots \overline{Q}_n z_n\, \neg \ell^x_\Delta[\ell^y_\Gamma[\mathrm{PI}(\pi)]]$ where $\overline{Q}_i = \exists$ ($\forall$) if $Q_i = \forall$ ($\exists$). Therefore also $\Delta \models \neg Q_1 z_1 \ldots Q_n z_n\, \ell^x_\Delta[\ell^y_\Gamma[\mathrm{PI}(\pi)]]$. By Lemma 4.12 and as all variables which replace colored terms are bound, $\Delta \models \neg Q_1 z_1 \ldots Q_n z_n\, \ell^y_\Gamma[\ell^x_\Delta[\mathrm{PI}(\pi)]]$.

As it is now easy to see that $Q_1 z_1 \ldots Q_n z_n\, \ell^y_\Gamma[\ell^x_\Delta[\mathrm{PI}(\pi)]]$ contains no colored symbol, it is an interpolant. □

Appendix **A**

WT: Huang's proof

A.1 Propositional interpolants

Let $\Gamma \cup \Delta$ be unsatisfiable. Let π be a proof of the empty clause from $\Gamma \cup \Delta$. Then PI is a function that returns a interpolant with respect to the current clause.

Definition A.1 (Propositional interpolant)**.** Let π be a resolution refutation of $\Gamma \cup \Delta$. A formula A is a *propositional interpolant* if

1. $\Gamma \models A$
2. $\Delta \models \neg A$
3. $\mathrm{PS}(A) \subseteq (\mathrm{PS}(\Gamma) \cap \mathrm{PS}(\Delta)) \cup \{\top, \bot\}$.

For a clause C in π, a formula A_C is a *propositional interpolant relative to* C if

1. $\Gamma \models A_C \vee C$
2. $\Delta \models \neg A_C \vee C$
3. $\mathrm{PS}(A_C) \subseteq (\mathrm{PS}(\Gamma) \cap \mathrm{PS}(\Delta)) \cup \{\top, \bot\}$.

The propositional interpolant of the empty clause derived in π is denoted by $\mathrm{PI}(\pi)$. △

The third condition of a propositional interpolant will sometimes be referred to as *language restriction*. It is easy to see that the propositional interpolant relative to the empty clause of a resolution refutation is a propositional interpolant.

We proceed by defining a procedure PI which extracts propositional interpolants from a resolution refutation.

Definition A.2 (Propositional interpolant extraction.)**.** Let π be a resolution refutation of $\Gamma \cup \Delta$. $\mathrm{PI}(\pi)$ is defined to be $\mathrm{PI}(\square)$, where $\square$ is the empty clause derived in π.

For a clause C in π, $\mathrm{PI}(C)$ is defined as follows:

Base case. If $C \in \Gamma$, $\mathrm{PI}(C) = \bot$. If otherwise $C \in \Delta$, $\mathrm{PI}(C) = \top$.

Resolution. If the clause C is the result of a resolution step of $C_1 : D \vee l$ and $C_2 : E \vee \neg l'$ using a unifier σ such that $l\sigma = l'\sigma$, then $\mathrm{PI}(C)$ is defined as follows:

1. If l is Γ-colored: $\mathrm{PI}(C) = [\mathrm{PI}(C_1) \vee \mathrm{PI}(C_2)]\sigma$
2. If l is Δ-colored: $\mathrm{PI}(C) = [\mathrm{PI}(C_1) \wedge \mathrm{PI}(C_2)]\sigma$
3. If l is grey: $\mathrm{PI}(C) = [(l \wedge \mathrm{PI}(C_2)) \vee (\neg l' \wedge \mathrm{PI}(C_1))]\sigma$

Factorisation. If the clause C is the result of a factorisation of $C_1 : l \vee l' \vee D$ using a unifier σ such that $l\sigma = l'\sigma$, then $\mathrm{PI}(C) = \mathrm{PI}(C_1)\sigma$.

Paramodulation. Suppose the clause C is the result of a paramodulation of $C_1 : s = t \vee C$ and $C_2 : D[r]$ using a unifier σ such that $r\sigma = s\sigma$. Let $h[r]$ be the maximal colored term in which r occurs in $D[r]$. Then $\mathrm{PI}(C)$ is defined according to the following case distinction:

1. If $h[r]$ is Δ-colored and $h[r]$ occurs more than once in $D[r] \vee \mathrm{PI}(D[r])$:
   $\mathrm{PI}(C) = [(s = t \wedge \mathrm{PI}(C_2)) \vee (s \neq t \wedge \mathrm{PI}(C_1))]\sigma \vee (s = t \wedge h[s] \neq h[t])\sigma$
2. If $h[r]$ is Γ-colored and $h[r]$ occurs more than once in $D[r] \vee \mathrm{PI}(D[r])$:
   $\mathrm{PI}(C) = [(s = t \wedge \mathrm{PI}(C_2)) \vee (s \neq t \wedge \mathrm{PI}(C_1))]\sigma \wedge (s \neq t \vee h[s] = h[t])\sigma$
3. Otherwise:
   $\mathrm{PI}(C) = [(s = t \wedge \mathrm{PI}(C_2)) \vee (s \neq t \wedge \mathrm{PI}(C_1))]\sigma$ $\triangle$

Proposition A.3. *Let C be a clause of a resolution refutation of $\Gamma \cup \Delta$. Then $\mathrm{PI}(C)$ is a propositional interpolant with respect to C.*

Proof. Proof by induction on the number of rule applications including the following strengthenings: $\Gamma \models \mathrm{PI}(C) \vee C_\Gamma$ and $\Delta \models \neg\,\mathrm{PI}(C) \vee C_\Delta$, where D_Φ denotes the clause D with only the literals which are contained in $\mathrm{L}(\Phi)$. They clearly imply conditions 1 and 2 of definition A.1.

Base case. Suppose no rules were applied. We distinguish two possible cases:

1. $C \in \Gamma$. Then $\mathrm{PI}(C) = \bot$. Clearly $\Gamma \models \bot \vee C_\Gamma$ as $C_\Gamma = C \in \Gamma$, $\Delta \models \neg\bot \vee C_\Delta$ and $\bot$ satisfies the restriction on the language.
2. $C \in \Delta$. Then $\mathrm{PI}(C) = \top$. Clearly $\Gamma \models \top \vee C_\Gamma$, $\Delta \models \neg\top \vee C_\Delta$ as $C_\Delta = C \in \Delta$ and $\top$ satisfies the restriction on the language.

Suppose the property holds for n rule applications. We show that it holds for $n+1$ applications by considering the last one:

Resolution. Suppose the last rule application is an instance of resolution. Then it is of the form:

$$\frac{C_1 : D \vee l \qquad C_2 : E \vee \neg l'}{C : (D \vee E)\sigma} \quad l\sigma = l'\sigma$$

By the induction hypothesis, we can assume that:

$\Gamma \models \mathrm{PI}(C_1) \vee (D \vee l)_\Gamma$

$\Delta \models \neg\,\mathrm{PI}(C_1) \vee (D \vee l)_\Delta$

$\Gamma \models \mathrm{PI}(C_2) \vee (E \vee \neg l')_\Gamma$

$\Delta \models \neg\,\mathrm{PI}(C_2) \vee (E \vee \neg l')_\Delta$

We consider the respective cases from definition A.2:

1. l is Γ-colored. Then $\mathrm{PI}(C) = [\mathrm{PI}(C_1) \vee \mathrm{PI}(C_2)]\sigma$.
   As $\mathrm{PS}(l) \in \mathrm{L}(\Gamma)$, $\Gamma \models (\mathrm{PI}(C_1) \vee D_\Gamma \vee l)\sigma$ as well as $\Gamma \models (\mathrm{PI}(C_2) \vee E_\Gamma \vee \neg l')\sigma$. By a resolution step, we get $\Gamma \models (\mathrm{PI}(C_1) \vee \mathrm{PI}(C_2))\sigma \vee ((D \vee E)\sigma)_\Gamma$.
   Furthermore, as $\mathrm{PS}(l) \notin \mathrm{L}(\mathrm{PI})$, $\Delta \models (\neg\,\mathrm{PI}(C_1) \vee D_\Delta)\sigma$ as well as $\Delta \models (\neg\,\mathrm{PI}(C_2) \vee E_\Delta)\sigma$. Hence it certainly holds that $\Delta \models (\neg\,\mathrm{PI}(C_1) \vee \neg\,\mathrm{PI}(C_2))\sigma \vee (D \vee E)\sigma_\Delta$.
   The language restriction clearly remains satisfied as no non-logical symbols are added.
2. l is Δ-colored. Then $\mathrm{PI}(C) = [\mathrm{PI}(C_1) \wedge \mathrm{PI}(C_2)]\sigma$.
   As $\mathrm{PS}(l) \notin \mathrm{L}(\Gamma)$, $\Gamma \models (\mathrm{PI}(C_1) \vee D_\Gamma)\sigma$ as well as $\Gamma \models (\mathrm{PI}(C_2) \vee E_\Gamma)\sigma$. Suppose that in a model M of Γ, $M \not\models D_\Gamma$ and $M \not\models E_\Gamma$. Then $M \models \mathrm{PI}(C_1) \wedge \mathrm{PI}(C_2)$. Hence $\Gamma \models (\mathrm{PI}(C_1) \wedge \mathrm{PI}(C_2))\sigma \vee ((D \vee E)\sigma)_\Gamma$.
   Furthermore due to $\mathrm{PS}(l) \in \mathrm{L}(\Delta)$, $\Delta \models (\neg\,\mathrm{PI}(C_1) \vee D_\Delta \vee l)\sigma$ as well as $\Delta \models (\neg\,\mathrm{PI}(C_2) \vee E_\Delta \vee \neg l')\sigma$. By a resolution step, we get $\Delta \models (\neg\,\mathrm{PI}(C_1) \vee \neg\,\mathrm{PI}(C_2))\sigma \vee (D_\Delta \vee E_\Delta)\sigma$ and hence $\Delta \models \neg(\mathrm{PI}(C_1) \wedge \mathrm{PI}(C_2))\sigma \vee (D_\Delta \vee E_\Delta)\sigma$.
   The language restriction again remains intact.
3. l is grey. Then $\mathrm{PI}(C) = [(l \wedge \mathrm{PI}(C_2)) \vee (\neg l' \wedge \mathrm{PI}(C_1))]\sigma$
   First, we have to show that $\Gamma \models [(l \wedge \mathrm{PI}(C_2)) \vee (l' \wedge \mathrm{PI}(C_1))]\sigma \vee ((D \vee E)\sigma)_\Gamma$. Suppose that in a model M of Γ, $M \not\models D_\Gamma$ and $\Gamma \not\models E$. Otherwise we are done. The induction assumption hence simplifies to $M \models \mathrm{PI}(C_1) \vee l$ and $M \models \mathrm{PI}(C_2) \vee \neg l'$ respectively. As $l\sigma = l'\sigma$, by a case distinction argument on the truth value of $l\sigma$, we get that either $M \models (l \wedge \mathrm{PI}(C_2))\sigma$ or $M \models (\neg l' \wedge \mathrm{PI}(C_1))\sigma$.
   Second, we show that $\Delta \models ((l \vee \neg\,\mathrm{PI}(C_1)) \wedge (\neg l' \vee \neg\,\mathrm{PI}(C_2)))\sigma \vee ((D \vee E)\sigma)_\Delta$. Suppose again that in a model M of Δ, $M \not\models D_\Delta$ and $\Gamma \not\models E_\Delta$. Then the required statement follows from the induction hypothesis.
   The language condition remains satisfied as only the common literal l is added to the interpolant.

Factorisation. Suppose the last rule application is an instance of factorisation. Then it is of the form:

$$\frac{C_1 : l \vee l' \vee D}{C : (l \vee D)\sigma} \quad \sigma = \mathrm{mgu}(l, l')$$

Then the propositional interpolant $\mathrm{PI}(C)$ is defined as $\mathrm{PI}(C_1)$. By the induction hypothesis, we have:

$\Gamma \models \mathrm{PI}(C_1) \vee (l \vee l' \vee D)_\Gamma$

$\Delta \models \mathrm{PI}(C_1) \vee (l \vee l' \vee D)_\Delta$

It is easy to see that then also:

$\Gamma \models (\mathrm{PI}(C_1) \vee (l \vee D)_\Gamma)\sigma$

$\Delta \models (\mathrm{PI}(C_1)\sigma \vee (l \vee D)_\Delta)\sigma$

The restriction on the language trivially remains intact.

Paramodulation. Suppose the last rule application is an instance of paramodulation. Then it is of the form:

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

By the induction hypothesis, we have:

$\Gamma \models \mathrm{PI}(C_1) \vee (D \vee s = t)_\Gamma$

$\Delta \models \neg\, \mathrm{PI}(C_1) \vee (D \vee s = t)_\Delta$

$\Gamma \models \mathrm{PI}(C_2) \vee (E[r])_\Gamma$

$\Delta \models \neg\, \mathrm{PI}(C_2) \vee (E[r])_\Delta$

First, we show that $\mathrm{PI}(C)$ as constructed in case 3 of the definition is a propositional interpolant in any of these cases:

$\mathrm{PI}(C) = (s = t \wedge \mathrm{PI}(C_2)) \vee (s \neq t \wedge \mathrm{PI}(C_1))$

Suppose that in a model M of Γ, $M \not\models D\sigma$ and $M \not\models E[t]_p\sigma$. Otherwise we are done. Furthermore, assume that $M \models (s = t)\sigma$. Then $M \not\models E[r]_p\sigma$, but then necessarily $M \models \mathrm{PI}(C_2)\sigma$.
On the other hand, suppose $M \models (s \neq t)\sigma$. As also $M \not\models D\sigma$, $M \models \mathrm{PI}(C_1)\sigma$. Consequently, $M \models [(s = t \wedge \mathrm{PI}(C_2)) \vee (s \neq t \wedge \mathrm{PI}(C_1))]\sigma \vee [(D \vee E)_\Gamma]\sigma$

By an analogous argument, we get $\Delta \models [(s = t \wedge \neg\, \mathrm{PI}(C_2)) \vee (s \neq t \wedge \neg\, \mathrm{PI}(C_1))]\sigma \vee [(D \vee E)_\Delta]\sigma$, which implies $\Delta \models [(s \neq t \vee \neg\, \mathrm{PI}(C_2)) \wedge (s = t \vee \neg\, \mathrm{PI}(C_1))]\sigma \vee ((D \vee E)_\Delta)\sigma$

The language restriction again remains satisfied as the only predicate, that is added to the interpolant, is $=$.

This concludes the argumentation for case 3.

The interpolant for case 1 differs only by an additional formula added via a disjunction and hence condition 1 of definition A.1 holds by the above reasoning. As the

adjoined formula is a contradiction, its negation is valid which in combination with the above reasoning establishes condition 2. Since no new predicated are added, the language condition remains intact.

The situation in case 2 is somewhat symmetric: As a tautology is added to the interpolant with respect to case 1, condition 1 is satisfied by the above reasoning. For condition 2, consider that the negated interpolant for case 1 implies the negated interpolant for this case. The language condition again remains intact. □

A.2 Propositional refutations

Before we are able to specify a procedure to transform the propositional interpolant generated by PI into a proper interpolant without any colored terms, we need to make some observations about tree refutations.

In a tree refutation where the input clauses have a disjoint sets of variables, every variable has a unique ancestor which traces back to an input clause and hence appears only along a certain path. This insight allows us to push substitutions of the variables upwards along this path and arrive at the following definition and lemma:

Definition A.4. A resolution refutation is a *propositional refutation* if no nontrivial substitutions are employed. △

Lemma A.5. *Let Φ be unsatisfiable. Then there is a propositional refutation of Φ which starts from instances of Φ.*

Proof. Let π be a resolution refutation of Φ. By Lemma 2.4, we can assume without loss of generality that π is a tree refutation where the sets of variables of the input clauses are disjoint. Furthermore, we can assume that only most general unifiers are employed in π.

Then any unifier in π is either trivial on x or there is one unique unifier σ in π with $x\sigma = t$ where x does not occur in t. Hence along the path through the deduction where x occurs, it remains unchanged. Therefore we can create a new resolution refutation π' from π where x is replaced by t. Clearly π' is rooted in instances of Φ.

By application of this procedure to all variable occurring in π, we obtain a desired resolution refutation. □

Even though propositional refutations have nice properties for theoretical analysis, their use in practise is not desired as its construction involves a considerable blowup of the refutation. But its use is still justified in this instance as we can show for arbitrary refutations π that the algorithm stated in A.2 gives closely related results for both π and its corresponding propositional refutation.

Lemma A.6. *Let π be a resolution refutation of Φ and π' a propositional refutation corresponding to π. Then for every clause C in π and its corresponding clause C' in π', $\mathrm{PI}(C)\sigma = \mathrm{PI}(C')$, where σ is the composition of the unifications of π which are applied to the variables occurring in C .*

Proof. For the construction of the propositional skeleton of PI(·) only the coloring of the clauses is relevant and since this is the same in both π and π', it coincides for PI(C) and PI(C').

Hence PI(C) and PI(C') differ only in their term structure. To be more specific, in PI(C'), the composition of substitutions that are applied in π have already been applied to the initial clauses of π'. Note that substitution commutes with the rules of resolution. Therefore the only difference between PI(C) and PI(C') is that at certain term positions, there are variables in PI(C) where in PI(C') by some substitution a different term is located. But these substitutions are certainly applied by σ, hence $\mathrm{PI}(C)\sigma = \mathrm{PI}(C')$. □

this part occurs in other proof

A.3 Lifting of colored symbols

This establishes the theoretical framework which is required to define and show the correctness of the procedure to construct a proper interpolant from the propositional interpolant. The idea of this procedure will be to replace colored terms still occurring in the propositional interpolant with variables and quantifying them appropriately. This replacement is referred to as lifting:

Definition A.7 (Lifting)**.** Let Γ and Δ be sets of first-order formulas, ϕ a formula or a term, $Z = \{\zeta_1, \ldots, \zeta_n\}$ the maximal Φ-terms for $\Phi \in \{\Gamma, \Delta\}$ in ϕ and $z_1, \ldots, z_n$ fresh variables, referred to as *lifting variables*.

Then $\ell^z_\Phi[\phi]$ denotes $\ell^z_\Phi[\phi, Z]$ which is defined as follows:

$$\ell^z_\Phi[\phi, Z] \stackrel{\text{def}}{=} \begin{cases} \phi & Z = \varnothing \\ \ell^z_\Phi[\phi\{\zeta_i/z_i\}, Z\backslash\{\zeta_i\}] & \zeta_i \in Z \text{ such that } \zeta_i \text{ is not subterm of another term in } Z \end{cases}$$

We also denote ϕ lifted of both Γ- and Δ-terms by $\ell[\phi]$ if the variables replacing the colored terms are clear from the context or are not crucial. △

We usually lift Δ-terms by indexed variables with letter x and Γ-terms with letter y. If the lifting is not specific to a color, we use indexed variables with letter z.

Some elementary properties of liftings are described by the following lemmas:

Lemma A.8. *Let A and B be first-order formulas and s and t be terms. Then it holds that:*

1. $\ell^z_\Phi[\neg A] \Leftrightarrow \neg \ell^z_\Phi[A]$
2. $\ell^z_\Phi[A \circ B] \Leftrightarrow (\ell^z_\Phi[A] \circ \ell^z_\Phi[B])$ *for* $\circ \in \{\wedge, \vee\}$
3. $\ell^z_\Phi[s = t] \Leftrightarrow (\ell^z_\Phi[s] = \ell^z_\Phi[t])$

Lemma A.9. $\ell^y_\Gamma[\ell^x_\Delta[C]]$ *and* $\ell^{x'}_\Delta[\ell^{y'}_\Gamma[C]]$ *differ only in the naming of the variables replacing maximal colored terms.*

Proof. Suppose a term t in C is affected by a lifting. We only need to consider maximal colored terms as grey terms are not affected by the liftings. Without loss of generality let t be a maximal Δ-colored term.

Let Φ be the positions of maximal occurrences of t. Then in the left hand side, exactly all terms at positions Φ are replaced by x_i for some i.

In the right hand side, all terms at positions Φ are replaced by $\ell^{y'}_\Gamma[t]$ first. However after this step, all these terms are equal to $\ell^{y'}_\Gamma[t]$, and as all distinct maximal Γ-terms are replaced by distinct variables, no other maximal colored term is equal to $\ell^{y'}_\Gamma[t]$. Hence exactly the terms at positions Φ are replaced by the same variable x'_j for some j. □

First, we consider the lifting of the Δ-terms:

Lemma A.10. *Let* π *be a resolution refutation of* $\Gamma \cup \Delta$. *Then* $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C) \vee C]$ *for* C *in* π.

Proof. We proof this result by induction on the number of rule applications in the propositional refutation corresponding to π. Similar to the proof of A.3, we show the strengthening: $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C) \vee C_\Gamma]$ for C in π.

Base case. If no rules have been applied, C is an instance of a clause of either Γ or Δ. In the former case, all Δ-terms of C were added by unification, hence by replacing them with variables, we obtain a clause C' which still is an instance of C and consequently is implied by Γ. In the latter case, $\mathrm{PI}(C) = \top$.

Resolution. Suppose the last rule application is an instance of resolution. Then it is of the form:

$$\frac{C_1 : D \vee l \qquad C_2 : E \vee \neg l}{C : D \vee E}$$

By the induction hypothesis,

$\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1) \vee (D \vee l)_\Gamma]$ and

$\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_2) \vee (E \vee \neg l)_\Gamma]$

which by Lemma A.8 is equivalent to

$\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1)] \vee \ell^x_\Delta[D_\Gamma] \vee \ell^x_\Delta[l_\Gamma]$ $^{(\circ)}$ and

$\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_2)] \vee \ell^x_\Delta[E_\Gamma] \vee \neg\ell^x_\Delta[l_\Gamma]$ $^{(*)}$.

1. Suppose l is Γ-colored. Then $\mathrm{PI}(C) = \mathrm{PI}(C_1) \vee \mathrm{PI}(C_2)$. By using resolution of $^{(*)}$ and $^{(\circ)}$ on $\ell^x_\Delta[l_\Gamma]$, we get that

$$\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1)] \vee \ell^x_\Delta[\mathrm{PI}(C_2)] \vee \ell^x_\Delta[D_\Gamma] \vee \ell^x_\Delta[E_\Gamma].$$

Several applications of Lemma A.8 give $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1) \vee \mathrm{PI}(C_2) \vee (D \vee E)_\Gamma]$.

2. Suppose l is Δ-colored. Then $\mathrm{PI}(C) = \mathrm{PI}(C_1) \wedge \mathrm{PI}(C_2)$.
   As l and $\neg l$ are not contained in $\mathrm{L}(\Gamma)$, we get that
   $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1)] \vee \ell^x_\Delta[D_\Gamma]$ and
   $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_2)] \vee \ell^x_\Delta[E_\Gamma]$.
   So if in a model M of Γ we have that $M \not\models \ell^x_\Delta[D_\Gamma]$ and $M \not\models \ell^x_\Delta[E_\Gamma]$, it follows that $M \models \ell^x_\Delta[\mathrm{PI}(C_1)]$ and $M \models \ell^x_\Delta[\mathrm{PI}(C_2)]$. Hence by Lemma A.8 $M \models \ell^x_\Delta[\mathrm{PI}(C_1) \wedge \mathrm{PI}(C_2)] \vee \ell^x_\Delta[(D \vee E)_\Gamma]$.
3. Suppose l is grey. Then $\mathrm{PI}(C) = (l \wedge \mathrm{PI}(C_2)) \vee (\neg l \wedge \mathrm{PI}(C_1))$.
   We show that $\Gamma \models \ell^x_\Delta[(l \wedge \mathrm{PI}(C_2)) \vee (\neg l \wedge \mathrm{PI}(C_1)) \vee (D \vee E)_\Gamma]$.
   Suppose that for a model M of Γ that $M \not\models \ell^x_\Delta[D_\Gamma]$ and $M \not\models \ell^x_\Delta[E_\Gamma]$. Then by $^{(\circ)}$ and $^{(*)}$, we get that
   $M \models \ell^x_\Delta[\mathrm{PI}(C_1)] \vee \ell^x_\Delta[l_\Gamma]$ as well as
   $M \models \ell^x_\Delta[\mathrm{PI}(C_2)] \vee \neg\ell^x_\Delta[l_\Gamma]$.
   So $M \models \ell^x_\Delta[l_\Gamma]$ implies that $M \models \ell^x_\Delta[\mathrm{PI}(C_2)]$ and $M \models \neg\ell^x_\Delta[l_\Gamma]$ implies that $M \models \ell^x_\Delta[\mathrm{PI}(C_1)]$ and
   Therefore $M \models (\ell^x_\Delta[l] \wedge \ell^x_\Delta[\mathrm{PI}(C_2)]) \vee (\neg\ell^x_\Delta[l] \wedge \ell^x_\Delta[\mathrm{PI}(C_1)]) \vee (\ell^x_\Delta[D_\Gamma] \vee \ell^x_\Delta[E_\Gamma])$, and several applications of Lemma A.8 give $M \models \ell^x_\Delta[(l \wedge \mathrm{PI}(C_2)) \vee (\neg l \wedge \mathrm{PI}(C_1)) \vee (D_\Gamma \vee E_\Gamma)]$.

Factorisation. Suppose the last rule application is an instance of factorisation. Then it is of the form:

$$\frac{C_1 : l \vee l \vee D}{C : l \vee D}$$

The propositional interpolant directly carried over from C_1, i.e. $\mathrm{PI}(C) = \mathrm{PI}(C_1)$.

By the induction hypothesis, we get that $\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1) \vee (l \vee l \vee D)_\Gamma]$. By Lemma A.8,

$\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1)] \vee (\ell^x_\Delta[l_\Gamma] \vee \ell^x_\Delta[l_\Gamma] \vee \ell^x_\Delta[D_\Gamma])$,

which clearly is equivalent to

$\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1)] \vee (\ell^x_\Delta[l_\Gamma] \vee \ell^x_\Delta[D_\Gamma])$,

so by again applying Lemma A.8, we arrive at

$\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1) \vee (l \vee D)_\Gamma]$.

Paramodulation. Suppose the last rule application is an instance of paramodulation. Then it is of the form:

$$\frac{C_1 : D \vee s = t \qquad C_2 : E[s]_p}{C : D \vee E[t]_p}$$

By the induction hypothesis, we have that

$\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1) \vee (D \vee s = t)_\Gamma]$ and

$\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_2) \vee (E[s]_p)_\Gamma]$.

By Lemma A.8, we get that

$\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_1)] \vee \ell^x_\Delta[D_\Gamma] \vee \ell^x_\Delta[s] = \ell^x_\Delta[t]$ and

$\Gamma \models \ell^x_\Delta[\mathrm{PI}(C_2)] \vee \ell^x_\Delta[(E[s]_p)_\Gamma]$.

We distinguish two cases:

1. Suppose s does not occur in a maximal Δ-term $h[s]$ in $E[s]_p$ which occurs more than once in $\mathrm{PI}(E(s)) \vee E[s]_p$.

   We show that $\Gamma \models \ell^x_\Delta[(s = t \wedge \mathrm{PI}(C_2)) \vee (s \neq t \wedge \mathrm{PI}(C_1)) \vee (D \vee E[t]_p)_\Gamma]$, which subsumes the cases 2 and 3 of Definition A.2. By Lemma A.8, this is equivalent to

   $\Gamma \models (\ell^x_\Delta[s] = \ell^x_\Delta[t] \wedge \ell^x_\Delta[\mathrm{PI}(C_2)]) \vee (\ell^x_\Delta[s] \neq \ell^x_\Delta[t] \wedge \ell^x_\Delta[\mathrm{PI}(C_1)]) \vee (\ell^x_\Delta[D_\Gamma] \vee \ell^x_\Delta[(E[t]_p)_\Gamma])$

   Suppose that in a model M of Γ, $M \not\models \ell^x_\Delta[D_\Gamma]$ and $M \not\models \ell^x_\Delta[(E[t]_p)_\Gamma]$. We show that then, depending on whether $\ell^x_\Delta[s] = \ell^x_\Delta[t]$ holds in M, one of the first two disjuncts holds in M.

   Then in case $M \models \ell^x_\Delta[s] = \ell^x_\Delta[t]$ we also get $M \not\models \ell^x_\Delta[(E[s]_p)_\Gamma]$ and consequently by the induction hypothesis $M \models \ell^x_\Delta[\mathrm{PI}(C_2)]$.

   However in case $M \models \ell^x_\Delta[s] \neq \ell^x_\Delta[t]$ we get by the induction hypothesis that $M \models \ell^x_\Delta[\mathrm{PI}(C_1)]$.

2. Otherwise s occurs in a maximal Δ-term $h[s]$ in $E[s]_p$ which occurs more than once in $\mathrm{PI}(E(s)) \vee E[s]_p$. This reflects case 1 of Definition A.2.

   Then models are possible in which $s = t$ and therefore $\ell^x_\Delta[s] = \ell^x_\Delta[t]$ holds, while at the same time $\ell^x_\Delta[h[s]] \neq \ell^x_\Delta[h[t]]$ does not as $h[s]$ and $h[t]$ are replaced by distinct variables due to being different Δ-terms.

   Therefore we amend the proof of case 1 as follows:

   In case $M \models \ell^x_\Delta[s] = \ell^x_\Delta[t]$ (otherwise proceed as in case 1), one of the following cases holds:

   - $M \models \ell^x_\Delta[h[s]] = \ell^x_\Delta[h[t]]$. From this, it follows that as in the proof of case 1, $M \not\models \ell^x_\Delta[(E[s]_p)_\Gamma]$ and consequently $M \models \ell^x_\Delta[\mathrm{PI}(C_2)]$ again by the induction hypothesis.
   - $M \models \ell^x_\Delta[h[s]] \neq \ell^x_\Delta[h[t]]$. However as here $\mathrm{PI}(C)$ contains the with respect to case 1 additional disjunct $s = t \wedge h[s] \neq h[t]$, $M \models \ell^x_\Delta[PI(C)]$ due to $M \models \ell^x_\Delta[s] = \ell^x_\Delta[t] \wedge \ell^x_\Delta[h[s]] \neq \ell^x_\Delta[h[t]]$ □

The interpolant extraction procedure PI exhibits a convenient property which is termed *symmetry* in [DKPW10, Definition 5] and can be stated formally as follows:

Lemma A.11. *Let π be a resolution refutation of $\Gamma \cup \Delta$ and $\hat{\pi}$ be π with $\hat{\Gamma} = \Delta$ and $\hat{\Delta} = \Gamma$. Then* $\mathrm{PI}(\pi) \Leftrightarrow \neg\, \mathrm{PI}(\hat{\pi})$.

Proof. We prove this lemma by induction on π. Let $\hat{\varphi}$ denote the clause/formula/literal/term in $\hat{\pi}$ corresponding to the clause/formula/literal/term φ in π.

Base case. If $C \in \Gamma$, then $C' \in \Delta'$ and $\mathrm{PI}(C) = \bot \Leftrightarrow \neg\top = \neg\, \mathrm{PI}(C')$. The case for $C \in \Delta$ is analogous.

Resolution. If the clause C is the result of a resolution step of $C_1 : D \vee l$ and $C_2 : E \vee \neg l'$ using a unifier σ such that $l\sigma = l'\sigma$, then by induction hypothesis, we get that $\mathrm{PI}(C_i) = \neg\, \mathrm{PI}(C_i')$ for $i \in \{1, 2\}$.

We distinguish the following cases:

1. l is Γ-colored. Then $\hat{l}$ is Δ-colored.

$$\begin{aligned}\mathrm{PI}(C) &= \mathrm{PI}(C_1) \vee \mathrm{PI}(C_2)\\ &\Leftrightarrow \neg(\neg\, \mathrm{PI}(C_1) \wedge \neg\, \mathrm{PI}(C_2))\\ &= \neg(\mathrm{PI}(\hat{C}_1) \wedge \mathrm{PI}(\hat{C}_2))\\ &= \neg\, \mathrm{PI}(\hat{C})\end{aligned}$$

2. l is Δ-colored. This case can be argued analogously.
3. l is grey. Then $\hat{l}$ is grey.

$$\begin{aligned}\mathrm{PI}(C) &= [(l \wedge \mathrm{PI}(C_2)) \vee (\neg l' \wedge \mathrm{PI}(C_1))]\sigma\\ &= (l\sigma \wedge \mathrm{PI}(C_2)\sigma) \vee (\neg l'\sigma \wedge \mathrm{PI}(C_1)\sigma)\\ &\Leftrightarrow (\neg l\sigma \vee \mathrm{PI}(C_2)\sigma) \wedge (l'\sigma \vee \mathrm{PI}(C_1)\sigma)\\ &\Leftrightarrow \neg[(l\sigma \wedge \neg\, \mathrm{PI}(C_2)\sigma) \vee (\neg l'\sigma \wedge \neg\, \mathrm{PI}(C_1)\sigma)]\\ &= \neg[(\hat{l}\sigma \wedge \neg\, \mathrm{PI}(C_2)\sigma) \vee (\neg \hat{l}'\sigma \wedge \neg\, \mathrm{PI}(C_1)\sigma)\\ &= \neg[(\hat{l} \wedge \neg\, \mathrm{PI}(C_2)) \vee (\neg \hat{l}' \wedge \neg\, \mathrm{PI}(C_1))]\sigma\\ &= \neg[(\hat{l} \wedge \mathrm{PI}(\hat{C}_2)) \vee (\neg \hat{l}' \wedge \mathrm{PI}(\hat{C}_1))]\sigma\\ &= \neg\, \mathrm{PI}(\hat{C})\end{aligned}$$

Factorisation. Suppose the clause C is the result of a factorisation of $C_1 : l \vee l' \vee D$. Then $\mathrm{PI}(C) = \mathrm{PI}(C_1)\sigma$ and the induction hypothesis gives the result.

Paramodulation. Suppose the clause C is the result of a paramodulation of $C_1 : s = t \vee C$ and $C_2 : D[r]$ using a unifier σ such that $r\sigma = s\sigma$. We distinguish the following cases:

1. r occurs in a maximal Δ-term $h[r]$ in $D[r]$ and $h[r]$ occurs more than once in $D[r] \vee \mathrm{PI}(D[r])$. Then $\hat{r}$ occurs in a maximal Γ-term $\hat{h}[r]$ in $\hat{D}[r]$ and $\hat{h}[r]$ occurs more than once in $\hat{D}[r] \vee \mathrm{PI}(\hat{D}[r])$.

$$\begin{aligned}
\mathrm{PI}(C) &= [(s = t \wedge \mathrm{PI}(C_2)) \vee (s \neq t \wedge \mathrm{PI}(C_1))]\sigma \vee (s = t \wedge h[s] \neq h[t])\sigma \\
&= [(s = t \wedge \neg\,\mathrm{PI}(\hat{C}_2)) \vee (s \neq t \wedge \neg\,\mathrm{PI}(\hat{C}_1))]\sigma \vee (s = t \wedge h[s] \neq h[t])\sigma \\
&\Leftrightarrow \neg[(s \neq t \vee \mathrm{PI}(\hat{C}_2)) \wedge (s = t \vee \mathrm{PI}(\hat{C}_1))]\sigma \wedge \neg(s \neq t \vee h[s] = h[t])\sigma \\
&\Leftrightarrow \neg[(s = t \wedge \mathrm{PI}(\hat{C}_2)) \vee (s \neq t \wedge \mathrm{PI}(\hat{C}_1))]\sigma \wedge \neg(s \neq t \vee h[s] = h[t])\sigma \\
&= \neg\,\mathrm{PI}(\hat{C})
\end{aligned}$$

2. r occurs in a maximal Γ-term $h(r)$ in $D[r]$ and $h(r)$ occurs more than once in $D[r] \vee \mathrm{PI}(D[r])$. This case can be argued analogously.
3. Otherwise:

$$\begin{aligned}
\mathrm{PI}(C) &= [(s = t \wedge \mathrm{PI}(C_2)) \vee (s \neq t \wedge \mathrm{PI}(C_1))]\sigma \\
&= [(s = t \wedge \neg\,\mathrm{PI}(\hat{C}_2)) \vee (s \neq t \wedge \neg\,\mathrm{PI}(\hat{C}_1))]\sigma \\
&\Leftrightarrow \neg[(s \neq t \vee \mathrm{PI}(\hat{C}_2)) \wedge (s = t \vee \mathrm{PI}(\hat{C}_1))]\sigma \\
&\Leftrightarrow \neg[(s = t \wedge \mathrm{PI}(\hat{C}_2)) \vee (s \neq t \wedge \mathrm{PI}(\hat{C}_1))]\sigma \\
&= \neg\,\mathrm{PI}(\hat{C})
\end{aligned}$$

□

This lemma can be leveraged to show a counterpart of Lemma A.10 for Δ:

Corollary A.12. *Let π be a resolution refutation of $\Gamma \cup \Delta$. Then $\Delta \models \ell^x_\Gamma[\neg\,\mathrm{PI}(C) \vee C]$ for C in π.*

Proof. Build $\hat{\pi}$ from π using $\hat{\Gamma} = \Delta$ and $\hat{\Delta} = \Gamma$ as initial clause set partition.
By Lemma A.10, $\hat{\Gamma} \models \ell^x_{\hat{\Delta}}[\mathrm{PI}(\hat{C}) \vee \hat{C}]$ for $\hat{C}$ in $\hat{\pi}$.
By Lemma A.11, $\hat{\Gamma} \models \ell^x_{\hat{\Delta}}[\neg\,\mathrm{PI}(C) \vee \hat{C}]$ for the clause C in π corresponding to $\hat{C}$ in $\hat{\pi}$.
This however is nothing else than $\Delta \models \ell^x_\Gamma[\neg\,\mathrm{PI}(C) \vee C]$. □

Theorem A.13. *Let π be a resolution refutation of $\Gamma \cup \Delta$ and $z_1, \ldots, z_n$ be the variables which replace the colored terms in $\ell^y_\Gamma[\ell^x_\Delta[\mathrm{PI}(\pi)]]$ ordered by their length. Then $Q_1 z_1 \ldots Q_n z_n\, \ell^y_\Gamma[\ell^x_\Delta[\mathrm{PI}(\pi)]]$, where Q_i is $\forall$ ($\exists$) if z_i replaces a Δ (Γ)-term, is an interpolant.*

Proof. By Lemma A.10, $\Gamma \models \forall x_1 \ldots \forall x_m\, \ell^x_\Delta[\mathrm{PI}(\pi)]$ where m is the number of maximal Δ-colored terms in $\mathrm{PI}(\pi)$.

A term in $\ell^x_\Delta[\mathrm{PI}(\pi)]$ is either x_i, $1 \leq i \leq m$, a grey term or a Γ-terms. Let t be a maximal Γ-term in $\ell^x_\Delta[\mathrm{PI}(\pi)]$ and $x_{j_1}, \ldots, x_{j_k}$ the variables replacing Δ-terms in t. Note that the Δ-terms, which are replaced by $x_{j_1}, \ldots, x_{i_{j_k}}$ respectively, are each of strictly smaller size than t as they are strict subterms of t.

In $\ell^y_\Gamma[\ell^x_\Delta[\mathrm{PI}(\pi)]]$, t is replaced by some z_j, which is existentially quantified. Hence t is a witness for z_j as due to the quantifier ordering, the existential quantification of z_j is in the scope of the quantifiers of $x_{j_1}, \ldots, x_{j_k}$ respectively. Therefore $\Gamma \models Q_1 z_1 \ldots Q_n z_n \, \ell^y_\Gamma[\ell^x_\Delta[\mathrm{PI}(\pi)]]$.

By Corollary A.12 $\Delta \models \forall y_1 \ldots \forall y_m \, \neg \ell^y_\Gamma[\mathrm{PI}(\pi)]$, where m is the number of Γ-colored terms in $\mathrm{PI}(\pi)$. By a similar line of argumentation as above, we can replace the maximal Δ-terms by existentially quantified variables and arrive at $\Delta \models \overline{Q}_1 z_1 \ldots \overline{Q}_n z_n \, \neg \ell^x_\Delta[\ell^y_\Gamma[\mathrm{PI}(\pi)]]$ where $\overline{Q}_i = \exists$ ($\forall$) if $Q_i = \forall$ ($\exists$). Therefore also $\Delta \models \neg Q_1 z_1 \ldots Q_n z_n \, \ell^x_\Delta[\ell^y_\Gamma[\mathrm{PI}(\pi)]]$. By Lemma A.9 and as all variables which replace colored terms are bound, $\Delta \models \neg Q_1 z_1 \ldots Q_n z_n \, \ell^y_\Gamma[\ell^x_\Delta[\mathrm{PI}(\pi)]]$.

As it is now easy to see that $Q_1 z_1 \ldots Q_n z_n \, \ell^y_\Gamma[\ell^x_\Delta[\mathrm{PI}(\pi)]]$ contains no colored symbol, it is an interpolant. □

Bibliography

[BJ13] Maria Paola Bonacina and Moa Johansson. On Interpolation in Automated Theorem Proving. Technical Report 86/2012, Dipartimento di Informatica, Università degli Studi di Verona, 2013. Submitted to journal August 2013.

[BL11] Matthias Baaz and Alexander Leitsch. *Methods of Cut-Elimination*. Trends in Logic. Springer, 2011.

[CK90] C.C. Chang and H.J. Keisler. *Model Theory*. Studies in Logic and the Foundations of Mathematics. Elsevier Science, 1990.

[Cra57a] William Craig. Linear Reasoning. A New Form of the Herbrand-Gentzen Theorem. *The Journal of Symbolic Logic*, 22(3):250–268, September 1957.

[Cra57b] William Craig. Three Uses of the Herbrand-Gentzen Theorem in Relating Model Theory and Proof Theory. *The Journal of Symbolic Logic*, 22(3):269–285, September 1957.

[DKPW10] Vijay D'Silva, Daniel Kroening, Mitra Purandare, and Georg Weissenbacher. Interpolant Strength. In *Proceedings of the International Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI)*, volume 5944 of *Lecture Notes in Computer Science*, pages 129–145. Springer, January 2010.

[Gen35] Gerhard Gentzen. Untersuchungen über das logische Schließen II. *Mathematische Zeitschrift*, 39, 1935.

[Hua95] Guoxiang Huang. Constructing Craig Interpolation Formulas. In *Proceedings of the First Annual International Conference on Computing and Combinatorics*, COCOON '95, pages 181–190, London, UK, UK, 1995. Springer-Verlag.

[Kle67] Stephen Cole Kleene. *Mathematical logic*. Wiley, New York, NY, 1967.

[Kra97] Jan Krajíček. Interpolation Theorems, Lower Bounds for Proof Systems, and Independence Results for Bounded Arithmetic. *Journal of Symbolic Logic*, pages 457–486, 1997.

[Lyn59] Roger C. Lyndon. An Interpolation Theorem in the Predicate Calculus. *Pacific Journal of Mathematics*, 9(1):129–142, 1959.

[McM03] Kenneth L. McMillan. Interpolation and SAT-Based Model Checking. In Jr. Hunt, Warren A. and Fabio Somenzi, editors, *Computer Aided Verification*, volume 2725 of *Lecture Notes in Computer Science*, pages 1–13. Springer Berlin Heidelberg, 2003.

[Pud97] Pavel Pudlák. Lower Bounds for Resolution and Cutting Plane Proofs and Monotone Computations. *J. Symb. Log.*, 62(3):981–998, 1997.

[Rob65] J. A. Robinson. A machine-oriented logic based on the resolution principle. *J. ACM*, 12(1):23–41, January 1965.

[Sho67] Joseph R. Shoenfield. *Mathematical logic*. Addison-Wesley series in logic. Addison-Wesley Pub. Co., 1967.

[Sla70] James R. Slagle. Interpolation theorems for resolution in lower predicate calculus. *J. ACM*, 17(3):535–542, July 1970.

[Tak87] Gaisi Takeuti. *Proof Theory*. Studies in logic and the foundations of mathematics. North-Holland, 1987.

[Wei10] Georg Weissenbacher. *Program Analysis with Interpolants*. PhD thesis, 2010.