Chapter 1

Interpolant extraction from resolution proofs in one phase

While the previous chapter demonstrates that it is possible to extract propositional interpolants and lift them from the colored symbols later in order to obtain a proper interpolant, we now present a novel approach, which only operates with grey intermediary interpolants. This is established by lifting any term which is added to the interpolant.

By its nature, this approach requires an alternate strategy than the proof of the extraction in two phases as a commutation of substitution and lifting is no longer possible if lifting variables are present. Let us recall the corresponding lemma from the previous chapter:

Lemma 1 (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 $\ell^z_\Phi[C\sigma] = \ell^z_\Phi[C]\sigma'$.

Consider the following illustration of a problem of the notion of applying this lemma to terms containing lifting variables:

Example 2. Let $\sigma = \{x \mapsto a\}$ and consider the terms $f(x)$ and $f(a)$, where f and a are colored symbols. Clearly $f(x)\sigma = f(a)$ and therefore necessarily $z_{f(x)}\sigma' = z_{f(a)}$.

But now consider $x_{f(x)}\sigma$. As $z_{f(x)}$ is a lifting variable, it is not affected by unifiers from resolution derivations and also not by σ. Hence $z_{f(x)}\sigma = z_{f(x)}$ and therefore $\ell[z_{f(x)}\sigma] = \ell[z_{f(x)}] = z_{f(x)}$, but $\ell[z_{f(x)}]\sigma' = z_{f(x)}\sigma' = z_{f(a)}$. So $\ell[z_{f(x)}\sigma] \neq \ell[z_{f(x)}]\sigma'$.

We see here that there are circumstances under which in order to commute lifting and substitution, the substitution σ' is required to conform to the equation $z_{f(x)}\sigma' = z_{f(a)}$, whereas in others, it must hold that $z_{f(x)}\sigma' = z_{f(x)}$. $\triangle$

1.1 Definition of the extraction algorithm

The extracted interpolants are prenex formulas, where the quantifier block and the matrix of the formula are calculated separately in each step of the traversal of the resolution refutation.

Extraction of the interpolant matrix $\mathrm{AI}_{\mathrm{mat}}$ and calculation of $\mathrm{AI}_{\mathrm{cl}}$

$\mathrm{AI}_{\mathrm{mat}}$ is inspired by the propositional interpolants PI from Definition **??**. Its difference lies in the fact that the lifting occurs in every step of the extraction. This however necessitates applying these liftings to the clauses of the resolution refutation as well. For a clause C of the resolution refutation, we will denote the clause with the respective liftings applied by $\mathrm{AI}_{\mathrm{cl}}(C)$ (a formal definition will be given below), and for a term t at position p in C, we denote $\mathrm{AI}_{\mathrm{cl}}(C)|_p$ by t_{AIcl}.

Now we can define preliminary versions of $\mathrm{AI}^{\bullet}_{\mathrm{mat}}$ and $\mathrm{AI}^{\bullet}_{\mathrm{cl}}$:

Definition 3 ($\mathrm{AI}^{\bullet}_{\mathrm{mat}}$ and $\mathrm{AI}^{\bullet}_{\mathrm{cl}}$)**.** Let π be a resolution refutation of $\Gamma \cup \Delta$.

For a clause C in π, $\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C)$ and $\mathrm{AI}^{\bullet}_{\mathrm{cl}}(C)$ are defined as follows:

Base case. If $C \in \Gamma$, $\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C) \overset{\mathrm{def}}{=} \bot$. If otherwise $C \in \Delta$, $\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C) \overset{\mathrm{def}}{=} \top$.

In any case, $\mathrm{AI}^{\bullet}_{\mathrm{cl}}(C) \overset{\mathrm{def}}{=} \ell[C]$.

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 $\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C)$ and $\mathrm{AI}^{\bullet}_{\mathrm{cl}}$ are defined as follows:

$\mathrm{AI}^{\bullet}_{\mathrm{cl}}(C) \overset{\mathrm{def}}{=} \ell[(\mathrm{AI}^{\bullet}_{\mathrm{cl}}(C_1) \setminus \{l_{\mathrm{AIcl}}\})\sigma] \ \vee \ \ell[(\mathrm{AI}^{\bullet}_{\mathrm{cl}}(C_2) \setminus \{l'_{\mathrm{AIcl}}\})\sigma]$

1. If l is Γ-colored: $\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C) \stackrel{\mathrm{def}}{=} \ell[\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C_1)\sigma] \vee \ell[\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C_2)\sigma]$
2. If l is Δ-colored: $\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C) \stackrel{\mathrm{def}}{=} \ell[\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C_1)\sigma] \wedge \ell[\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C_2)\sigma]$
3. If l is grey: $\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C) \stackrel{\mathrm{def}}{=} (\neg\ell[l'_{\mathrm{AIcl}}\sigma] \wedge \ell[\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C_1)\sigma]) \vee (\ell[l_{\mathrm{AIcl}}\sigma] \wedge \ell[\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C_2)\sigma])$

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 $\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C) \stackrel{\mathrm{def}}{=} \ell[\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C_1)\sigma]$ and $\mathrm{AI}^{\bullet}_{\mathrm{cl}}(C) \stackrel{\mathrm{def}}{=} \ell[(\mathrm{AI}_{\mathrm{cl}}(C_1) \setminus \{l'_{\mathrm{AIcl}}\})\sigma]$. $\triangle$

Note that in $\mathrm{AI}^{\bullet}_{\mathrm{mat}}$ and $\mathrm{AI}^{\bullet}_{\mathrm{cl}}$, it is possible that there for a colored term t in C that $t_{\mathrm{AIcl}} \neq z_t$ as illustrated by the following examples:

Example 4. We consider a resolution refutation of the initial clause sets $\Gamma = \{R(c), \neg Q(v)\}$ and $\Delta = \{\neg R(u) \vee Q(g(u))\}$:

$$\dfrac{\dfrac{R(c) \qquad \neg R(u) \vee Q(g(u))}{Q(g(c))}\ \mathrm{res}, y \mapsto c \qquad \neg Q(v)}{\Box}\ \mathrm{res}, v \mapsto g(c)$$

We now replace every clause C by $\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C) \mid \mathrm{AI}^{\bullet}_{\mathrm{cl}}(C)$ in order to visualise the steps of the algorithm:

$$\dfrac{\dfrac{\bot \mid R(y_c) \qquad \top \mid \neg R(u) \vee \neg Q(x_{g(u)})}{R(y_c) \mid Q(x_{g(u)})}\ \mathrm{res}, y \mapsto c \qquad \bot \mid \neg Q(v)}{\neg Q(x_{g(c)}) \wedge R(y_c) \mid \Box}\ \mathrm{res}, v \mapsto g(c)$$

By quantifying y_c existentially and $x_{g(c)}$ universally[1], we obtain an interpolant for $\Gamma \cup \Delta$: $\exists y_c \forall x_{g(c)} (\neg Q(x_{g_c}) \wedge R(y_c))$. Note however that $\ell[Q(g(c))] = Q(x_{g(c)})$, but $\mathrm{AI}_{\mathrm{mat}}(Q(g(c))) = Q(x_{g(u)})$. This example shows that this circumstance is not necessarily an obstacle for the correctness of this algorithm. $\triangle$

⟨exa:2b⟩ **Example 5.** We consider a resolution refutation of the initial clause sets $\Gamma = \{R(c), P(c)\}$ and $\Delta = \{\neg R(u) \vee \neg Q(g(u)), \neg P(v) \vee Q(g(v))\}$:

[1] The procedure for calculating the quantifier block is defined in section 1.1

$$\dfrac{\dfrac{\neg R(u) \vee \neg Q(g(u)) \qquad R(c)}{\neg Q(g(c))}\,\text{res}, u \mapsto c \qquad \dfrac{\neg P(v) \vee Q(g(v)) \qquad P(c)}{Q(g(c))}\,\text{res}, v \mapsto c}{\square}\,\text{res}$$

We now again display $\mathrm{AI}^{\bullet}_{\mathrm{mat}}(C) \mid \mathrm{AI}^{\bullet}_{\mathrm{cl}}(C)$ for every clause C of the refutation:

$$\dfrac{\dfrac{\top \mid \neg R(u) \vee \neg Q(x_{g(u)}) \qquad \bot \mid R(y_c)}{R(y_c) \mid \neg Q(x_{g(u)})}\,\text{res}, u \mapsto c \qquad \dfrac{\top \mid \neg P(v) \vee Q(x_{g(v)}) \qquad \bot \mid P(y_c)}{P(y_c) \mid Q(x_{g(v)})}\,\text{res}, v \mapsto c}{(Q(x_{g(v)}) \wedge R(y_c)) \ \vee \ (\neg Q(x_{g(u)}) \wedge P(y_c)) \mid \square}\,\text{res}$$

Note again that here, we have that $\ell[\neg Q(g(c))] = \neg Q(x_{g(c)}) \neq \mathrm{AI}^{\bullet}_{\mathrm{cl}}(\neg Q(g(c))) = \neg Q(x_{g(u)})$ and $\ell[Q(g(c))] = Q(x_{g(c)}) \neq \mathrm{AI}^{\bullet}_{\mathrm{cl}}(Q(g(c))) = Q(x_{g(v)})$. However in this instance, it is not possible to find quantifiers for the free variables of $\mathrm{AI}^{\bullet}_{\mathrm{mat}}(\square)$ such that by binding them, an interpolant is produced. For the naive approach, namely to use $\exists y_c \forall x_{g(v)} \forall x_{g(u)}$ as prefix, it holds that $\Gamma \not\models \exists y_c \forall x_{g(v)} \forall x_{g(u)}((Q(x_{g(v)}) \wedge R(y_c)) \ \vee \ (\neg Q(x_{g(u)}) \wedge P(y_c)))$. This failure is possible as intuitively, resolution deductions are valid by virtue of the resolved literals being equal. The interpolant extraction procedure exploits this property not directly on the clauses but on the lifted clause, i.e. on $\mathrm{AI}_{\mathrm{cl}}(C)$ for a clause C. Note that by ensuring that for resolved literals l and l', it holds that $l_{\mathrm{AIcl}} = l'_{\mathrm{AIcl}}$, we can obtain an interpolant, for instance: $\exists y_c \forall x^*((Q(x^*) \wedge R(y_c)) \ \vee \ (\neg Q(x^*) \wedge P(y_c)))$. △

In order to avoid the pitfall shown in Example 5 and to generalise the indicated solution, we define a function on resolved literals calculating a substitution, which ensures that the literals in the lifted clause, which correspond to the resolved literals, are equal.

Definition 6 (au)**.** For resolve literals l and l' of a resolution derivation step with a unifier σ such that $l\sigma = l'\sigma$, and $\ell[l_{\mathrm{AIcl}}\sigma] = P(s_1, \dots, s_n)$ and $\ell[l'_{\mathrm{AIcl}}\sigma] = P(t_1, \dots, t_n)$, we define:

$$\mathrm{au}(\ell[l_{\mathrm{AIcl}}\sigma], \ell[l'_{\mathrm{AIcl}}\sigma]) \stackrel{\mathrm{def}}{=} \bigcup_{i=1}^{n} \mathrm{au}(s_i, t_i)$$

For terms s and t where $s = \ell[l_{\mathrm{AIcl}}\sigma]|_p$ and $t = \ell[l'_{\mathrm{AIcl}}\sigma]|_p$ for some position p, we define:

$$\mathrm{au}(s,t) \stackrel{\mathrm{def}}{=} \begin{cases} \bigcup_{i=1}^{n} \mathrm{au}(s_i, t_i) & \text{if } s \text{ is grey and } s = f_s(s_1, \dots, s_n),\ t = f_t(t_1, \dots, t_n)^2 \\ \{z_{s'} \mapsto z_{\ell[s\sigma]}, z_{t'} \mapsto z_{\ell[s\sigma]}\} & \text{if } s \text{ is a lifting variable } z_{s'},\ t = z_{t'}, \text{ and } \ell[s\sigma] = \ell[t\sigma] = z_{\ell[s\sigma]} \end{cases} \quad \triangle$$

possibly argue here why au is well-defined (but it follows more or less directly from a later lemma)

Definition 7 ($\mathrm{AI}_{\mathrm{mat}}$ and $\mathrm{AI}_{\mathrm{cl}}$)**.** Let π be a resolution refutation of $\Gamma \cup \Delta$. $\mathrm{AI}_{\mathrm{mat}}(\pi)$ is defined to be $\mathrm{AI}_{\mathrm{mat}}(\Box)$, where $\Box$ is the empty box derived in π.

For a clause C in π, $\mathrm{AI}_{\mathrm{mat}}(C)$ and $\mathrm{AI}_{\mathrm{cl}}(C)$ are defined inductively as follows:

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

In any case, $\mathrm{AI}_{\mathrm{cl}}(C) \stackrel{\mathrm{def}}{=} \ell[C]$.

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 let $\tau = \mathrm{au}(\ell[l_{\mathrm{AIcl}}\sigma], \ell[l'_{\mathrm{AIcl}}\sigma])$ and define $\mathrm{AI}_{\mathrm{mat}}(C)$ and $\mathrm{AI}_{\mathrm{cl}}(C)$ as follows:

$$\mathrm{AI}_{\mathrm{cl}}(C) \stackrel{\mathrm{def}}{=} \ell[(\mathrm{AI}_{\mathrm{cl}}(C_1) \setminus \{l_{\mathrm{AIcl}}\})\sigma]\tau \ \vee\ \ell[(\mathrm{AI}_{\mathrm{cl}}(C_2) \setminus \{l'_{\mathrm{AIcl}}\})\sigma]\tau$$

1. If l is Γ-colored: $\mathrm{AI}_{\mathrm{mat}}(C) \stackrel{\mathrm{def}}{=} \ell[\mathrm{AI}_{\mathrm{mat}}(C_1)\sigma]\tau \vee \ell[\mathrm{AI}_{\mathrm{mat}}(C_2)\sigma]\tau$
2. If l is Δ-colored: $\mathrm{AI}_{\mathrm{mat}}(C) \stackrel{\mathrm{def}}{=} \ell[\mathrm{AI}_{\mathrm{mat}}(C_1)\sigma]\tau \wedge \ell[\mathrm{AI}_{\mathrm{mat}}(C_2)\sigma]\tau$
3. If l is grey: $\mathrm{AI}_{\mathrm{mat}}(C) \stackrel{\mathrm{def}}{=} (\neg\ell[l'_{\mathrm{AIcl}}\sigma]\tau \wedge \ell[\mathrm{AI}_{\mathrm{mat}}(C_1)\sigma]\tau) \ \vee\ (\ell[l_{\mathrm{AIcl}}\sigma]\tau \wedge \ell[\mathrm{AI}_{\mathrm{mat}}(C_2)\sigma]\tau)$

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:

$$\tau \stackrel{\mathrm{def}}{=} \mathrm{au}(\ell[l_{\mathrm{AIcl}}\sigma], \ell[l'_{\mathrm{AIcl}}\sigma]).$$

$$\mathrm{AI}_{\mathrm{mat}}(C) \stackrel{\mathrm{def}}{=} \ell[\mathrm{AI}_{\mathrm{mat}}(C_1)\sigma]\tau$$

$$\mathrm{AI}_{\mathrm{cl}}(C) \stackrel{\mathrm{def}}{=} \ell[(\mathrm{AI}_{\mathrm{cl}}(C_1) \setminus \{l'_{\mathrm{AIcl}}\})\sigma]\tau$$ △

Definition 8. $\mathrm{AI}^{\Delta}_{\mathrm{mat}}(C)$ ($\mathrm{AI}^{\Delta}_{\mathrm{cl}}(C)$) for a clause C is defined as $\mathrm{AI}_{\mathrm{mat}}(C)$ ($\mathrm{AI}_{\mathrm{cl}}(C)$) with the difference that in its inductive definition, every lifting $\ell[\varphi]$ for a formula or term φ is replaced by a lifting of only the Δ-terms $\ell_{\Delta}[\varphi]$. △

Lemma 9. *Let π be a resolution refutation of $\Gamma \cup \Delta$. Then for clauses C in π, $\Gamma \models \mathrm{AI}^{\Delta}_{\mathrm{mat}}(C) \vee \mathrm{AI}^{\Delta}_{\mathrm{cl}}(C)$.*

[2] Note that constants are treated as function symbols of arity zero.

Proof. We proceed by induction of the strengthening □

c:arrow_quantifier_block⟩

Theorem 10. *Let π be a resolution refutation of $\Gamma \cup \Delta$. Then $\mathrm{AI}_{\mathrm{mat}}(\pi)$ is an interpolant.*

Proof. This needs to many things I don't yet know how to make precise, so let's start with $\Gamma \models \ldots$

□