

1 Arrow-Algo

1. In the original clauses, find all occurrences of variables.

Common case: If a variable appears as outermost symbol or only has grey ancestor-terms, add an arrow from it to all other occurrences.

Uncommon case: if there is more than one occurrence of a variable under a Φ -colored term, add a *weak* dependency between them all (symmetric relation).

NOTE: this creates double arrows for occurrences at same depth. This appears to be necessary for terms which are only variables, and doesn't hurt if the variable is contained in a term.

2. For each step in the derivation:

- a) Build propositional interpolant using $\text{PI}(C_i)^*$, $i \in \{1, 2\}$, i.e. use ancestor PI without colored terms.
- b) If ancestors of atom added to $\text{PI}(C)$ had arrows, merge them to atom in $\text{PI}(C)$ (i.e. arrows starting in and leading to this atom).
- c) Replace colored terms in $\text{PI}(C)$ (from new atom and unifier applied to $\text{PI}(C_i)^*$) with fresh variables, except if a term has a double ended arrow to another overbinding variable, then use that variable.

An arrow starts (ends) in one of the new variables if it starts (ends) somewhere in the term it replaced.

- d) Collect quantifiers: from $\text{PI}(C_i)^*$, $i \in \{1, 2\}$ and ones from atom added to $\text{PI}(C)$. Order such that arrows only point to variables to the right AND weakly connected variables appear in the same quantifier block.

$$\bar{Q}_n = \text{sort}(Q_{n_1} \cup Q_{n_2} \cup \text{colored-terms}(l))$$

1.1 algo more formally

Every literal in any initial clause set has a globally unique id/number

Ex: $P(y, a, f(z, g(y, b))) \vee Q(x)$

Term position:

0.2.1.0 means first literal, 3rd arg, 2nd arg, 1st arg: y

0.1 is a

0.2.1 is $g(z, b)$

ρ calculates the position of a term or the term of a position, depending on the argument type.

for a position p_i , $\rho(p_i)$ denotes whatever p_i refers to in its respective clause.

for a term t , $\rho(t)$ denotes the position in t in its respective clause.

for a position p , $\rho_{\text{lit}}(p)$ denotes the position of the literal

for a position p , $\rho_{\text{term}}(p)$ denotes the position of the term in p_i

$\Rightarrow p = \rho_{\text{lit}}(p) \cdot \rho_{\text{term}}(p)$

for a position p , $p \bmod i$ denotes p with i least significant places cut off, $0.2.1.0 \bmod 2 = 0.2$

1.2 Arrows:

\mathcal{A} is a set of ordered pairs of term positions which point to positions in terms in literals

\mathcal{W} is a set of unordered pairs of term positions which point to positions in terms in literals

w.r.t a refutation π of $\Gamma \cup \Delta$:

1. For each initial clause C in $\Gamma \cup \Delta$:

Add to \mathcal{A} all (p_1, p_2) in C such that p_1 contains only grey symbol and $\rho(p_1)$ is a variable and $\rho(p_1) = \rho(p_2)$ but $p_1 \neq p_2$.

Add to \mathcal{W} all $\{p_1, p_2\}$ such that there is a colored symbol in p_1 and a possible different one in p_2 and $\rho(p_1)$ is a var and $\rho(p_1) = \rho(p_2)$.

2. For each C resulting from a resolution step from $C_1 : D \vee l$ and $C_2 : E \vee \neg l$ to $C = D \vee E$ with prop interpolant $\text{PI}(\cdot)$:

Note: literals are added to the interpolant if they occur in both ancestors.

Merge the respective ids of l and $\neg l$, i.e. their arrows. Their term structure will be the same, so all arrows point to valid positions.

1.3 algo

NOTE: for now, we assume that every colored-term has a globally unique id i and will be replaced by a variable with this index. This restriction is useful now and could potentially be lifted later, but it's not severe anyway.

Note: when a literal is added to the interpolant, the colored terms in one literal might have already been replaced with a certain variable before. we definitely have to use the same variable for both literals, and if one literal has other dependencies, we should stick with the variable we have.

PROBLEM: terms already replaced by variables still change! need to use same variable anyway, so note above not accurate!

1.3.1 AI_{mat} and AI_{cl}

Here, we define AI_{mat} , which represents the *matrix* of what will be the interpolant, and AI_{cl} , which represents the *clauses* in the refutation applied with the same unifications as AI_{mat} .

1. For each initial clause C , $\text{AI}_{\text{mat}}(C) = \text{PI}(C)$ and $\text{AI}_{\text{cl}}(C) = \ell[C]$.
2. For each C resulting from a resolution step from $C_1 : D \vee l$ and $C_2 : E \vee \neg l$ to $C = (D \vee E)\sigma$ with $l\sigma = l'\sigma$ with prop interpolant $\text{PI}(\cdot)$:

$l \dots$ literal in original clause

$l_{\text{cl}} \dots$ literal in AI_{cl} (with unifications and liftings carried over, such that ind hyp goes through)

$l\sigma = l'\sigma$, but l_{cl} and l'_{cl} might have been overbound with different variables. Still, they in a sense refer to the same ground literal, so we “can just” “unify” them.

Prose explanation of formal definition below: Shape must be the same in the sense that grey terms are the same, otherwise there is Φ -replacing-var vs Φ -replacing-var (let arbitrary one win) or Φ -replacing-var vs Φ -term (let var win) or Φ -term and Φ -term (replace both with same var). Also apply this to liftings in AI_{cl} and AI_{mat} here.

au is defined on terms which are used as parameters for literals l_{cl} and l'_{cl} which occur in $\text{AI}_{\text{cl}}(C_1)$, $\text{AI}_{\text{cl}}(C_2)$ such that for their corresponding l and l' , $l\sigma = l'\sigma$. Note that if one of the arguments of au has assigned a color, the other one either has none or the same color. There cannot be a conflict as otherwise their original form would not be unifiable. Note that $\text{au}(a, b)$ is well-defined, i.e. never maps

a variable to two different values as each occurrence of some x_j refers to a term with possible free variables, and since across the definition of au , always the same substitution σ is used as reference, every occurrence of x_j will be mapped to the same variable.

Let $l_{\text{cl}} = A(a_1, \dots, a_n)$, $l'_{\text{cl}} = A(b_1, \dots, b_n)$

$\text{au}(A(a_1, \dots, a_n), A(b_1, \dots, b_n)) = \bigcup_{i=1}^n \text{au}(a_i, b_i)$

$$\text{au}(a, b) = \begin{cases} \bigcup_{j=1}^n \text{au}(s_j, t_j) & \text{if } a = f_s(\bar{s}) \text{ grey and } b = f_t(\bar{t}) \text{ (includes } f_s \text{ being a constant)} \\ \{x_j \mapsto x_k\} & \text{if } a = x_j \text{ from lifting and } b \text{ a term with } \ell[b\sigma] = x_k \\ & \text{(note that } b \text{ is a variable as otherwise it would be colored and lifted)} \\ \{x_k \mapsto x_j\} & \text{if } b = x_k \text{ from lifting and } a \text{ a term with } \ell[a\sigma] = x_j \\ & \text{(note that } a \text{ is a variable as otherwise it would be colored and lifted)} \\ \{x_j \mapsto x_m, x_k \mapsto x_m\} & \text{if } a = x_j \text{ and } b = x_k, \text{ both from lifting, and } x_m \\ & \text{is the lifted term in the unified literal. More formally, } p = \rho(a) = \rho(b) \text{ and } \rho(\rho_{\text{lit}}(\ell[l\sigma]) \cdot \rho_{\text{term}}(p)) = \\ & \rho(\rho_{\text{lit}}(\ell[l'\sigma]) \cdot \rho_{\text{term}}(p)) = t_m \text{ and } \ell[t_m] = x_m. \\ \emptyset & \text{if } \ell[a\sigma] = \ell[b\sigma] = x_j \end{cases}$$

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

- if l and l' don't have the same color:

$$\text{AI}_{\text{mat}}(C) = (\neg \ell[l_{\text{cl}}\sigma] \tau \wedge \ell[\text{AI}_{\text{mat}}(C_1)\sigma] \tau) \vee (\ell[l_{\text{cl}}\sigma] \tau \wedge \ell[\text{AI}_{\text{mat}}(C_2)\sigma] \tau)$$

- if l and l' are Γ -colored :

$$\text{AI}_{\text{mat}}(C) = \ell[(\text{AI}_{\text{mat}}(C_1) \vee \text{AI}_{\text{mat}}(C_2)) \sigma] \tau$$

- if l and l' are Δ -colored:

$$\text{AI}_{\text{mat}}(C) = \ell[(\text{AI}_{\text{mat}}(C_1) \wedge \text{AI}_{\text{mat}}(C_2)) \sigma] \tau$$

1.3.2 AI

$$\text{AI}(C) = Q_1 u_1 \dots Q_m u_m \text{AI}_{\text{mat}}(C)$$

u_1, \dots, u_m are comprised of all x_i and y_i PLUS all free variables in $\text{AI}_{\text{mat}}(C)$.

Q_i is \exists if $u_i = y_i$ for some i , \forall if $u_i = x_i$ for some i . For free variables, Q_i as in C_1/C_2 .

if $u_i \sigma \neq u_i$, u_i becomes one of x_j or y_j or grey term.

$(p_1, p_2) \in \text{TransitiveClosure}(\mathcal{A})$ implies that $u_i < u_j$ if u_i replaces t_i and u_j replaces t_j and p_1 points into t_i and p_2 points into t_j .

2 proof of propositional aspect of AI

lift_subst_commute)

Lemma 1 (Restated from proof without propositional refutation, lemma 1). *Let C be a clause and σ a substitution. Let t_1, \dots, t_n be all maximal Δ -terms in this context, i.e. those that occur in C or $C\sigma$, and x_1, \dots, x_n the corresponding fresh variables to replace the t_i (i.e. none of the x_i occur in C). Define σ' such that for a variable z ,*

$$z\sigma' = \begin{cases} x_l & \text{if } z = x_k \text{ and } t_k\sigma = t_l \\ \ell_{\Delta,x}[z\sigma] & \text{otherwise} \end{cases}$$

Then $\ell_{\Delta,x}[C\sigma] = \ell_{\Delta,x}[C]\sigma'$.

Remark (Restriction of Lemma 1). Lemma 1 does not hold in case x_i occurs in C . This can easily be seen using the following counterexample:

Let $\sigma = \{x \mapsto a\}$ and $t_1 = f(x)$ and $t_2 = f(a)$. Then clearly $t_1\sigma = t_2$ and therefore $x_1\sigma' = x_2$.

But now consider $x_1\sigma$. As x_1 has its place in the domain of variables to replace colored terms, and σ is taken from a resolution refutation, they do not affect each other. Hence $x_1\sigma = x_1$ and therefore $\ell_{\Delta,x}[x_1\sigma] = x_1$, but $\ell_{\Delta,x}[x_1]\sigma' = \ell_{\Delta,x}[x_1]\sigma' = x_2$.

However such a situation arises naturally if we lift colored terms after every step of the interpolant extraction procedure, as there, the intermediate relative interpolants clearly contains variables to overbind terms, but we also need to treat terms that enter the interpolant by means of unification. \triangle

lift_logic_commute)

Lemma 2 (corresponds to Lemma 4.8 in thesis and Lemma 11 in Huang). *Let A and B be first-order formulas and s and t be terms. Then it holds that:*

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

a:no_colored_terms)

Lemma 3. $\text{AI}_{\text{mat}}(C)$ and $\text{AI}_{\text{cl}}(C)$ contain only grey terms and variables replacing colored terms. They do not contain colored terms. // true and used

?<corr:lift_ai>?

Corollary 4. *For a clause C in a resolution refutation π of $\Gamma \cup \Delta$:*

1. $\text{AI}_{\text{mat}}(C) = \ell[\text{AI}_{\text{mat}}(C)]$.
2. $\text{AI}_{\text{cl}}(C) = \ell[\text{AI}_{\text{cl}}(C)]$.

// true and unused

Lemma 5. *The x_i , y_i and z_i do not occur in any substitution of a resolution refutation.*

// true but unused

substitute_and_lift)

Lemma 6. *Let F be a formula without colored terms such that for a set of formulas Φ , $\Phi \models F$. Then $\Phi \models \ell[F\sigma]$ for a substitution σ .*

Proof. Note that substitutions only replace variables. Term positions, which are replaced by grey terms by σ are not affected by the lifting and hold due to being special cases of F .

Term positions, which are replaced by colored term by σ are again reduced to variables. All occurrences of a certain variable in F are substituted by the same term, so as the lifting replaces a certain term always be the same variables, all these occurrences of the variable are replaced by the same variable. \square

Lemma 7. *$\{x_j \mapsto x_k\} \in \text{au}(l_{\text{cl}}, l'_{\text{cl}})$ with $j \neq k$ implies that t_j (corresponding to x_j) contains a free variable.*

Proof. Let $\text{au}(s_{\text{cl}}, r_{\text{cl}})$ introduce $\{x_j \mapsto x_k\}$. We perform a case distinction:

- Suppose s_{cl} is a lifting variable and but r_{cl} is not. Then by the definition of au , $s_{\text{cl}} = x_j$ and $\ell[r_{\text{cl}}\sigma] = x_k$. Suppose that t_j does not have a free variable. Then in the resolution derivation, from the point on as it is lifted by x_j the original term does not change, hence $s = t_j$. As in consequence s does not contain free variables, $s\sigma = s$. As $s\sigma = r\sigma$ by the resolution rule application, we have that $s = r\sigma$. But as $\ell[s] = \ell[s_j] = x_j$, we must also have that $\ell[r\sigma] = x_j$. Hence $r\sigma = t_j$.

As r_{cl} occurs in $\text{AI}_{\text{cl}}(C)$ for some C , it is not a colored term, but as it is not a lifting variable and $r_{\text{cl}}\sigma$ is a colored term, r_{cl} must be a variable. By Lemma 11, as $r \in C$, r_{cl} is equal to s up to the index of lifting variables, hence $r = r_{\text{cl}}$.

But then $r_{\text{cl}}\sigma = r\sigma = t_j$, so $\ell[r_{\text{cl}}\sigma] = x_j$. But then $j = k$, a contradiction.

- Suppose r_{cl} is a lifting variable but s_{cl} is not. This case can be argued analogously.
- Suppose that both s_{cl} and r_{cl} are a lifting variables. Then by the definition of au , $s = r = t_k$ such that $\ell[t_k] = x_k$. Suppose that t_j does not contain a free variable. Then from the point on where s_{cl} has been lifted, t_j does not change. Therefore $s = t_j$. But then $k = j$, a contradiction.

If $\text{au}(s_{\text{cl}}, r_{\text{cl}})$ introduces $\{x_j \mapsto x_k\}$ and w.l.o.g. $s_{\text{cl}} = x_j$, then $\ell[r_{\text{cl}}\sigma] = x_k$ and there is a σ' such that $t_j\sigma' = s\sigma$.

But $s\sigma = r\sigma$, so $\ell[s\sigma] = \ell[r\sigma]$. Suppose that t_j does not have a free variable. As terms in resolution derivations are only changed by means of unification and t_j does not have free variables, it is invariant, so $t_j = s$. Hence $s\sigma = s$ and $t_j = r\sigma$

$$\ell[t_j] = \ell[s] = s_{\text{cl}} = x_j$$

If r_{cl} is a variable overbinding a colored term, then $r_{\text{cl}} = x_k$. TODO

Suppose r_{cl} is not a variable from an overbinding. As $\ell[r_{\text{cl}}\sigma] = x_k$,

$$\exists \sigma'' \text{ s.t. } \sigma'' = t_j$$

TODO

To show: $j = k$. □

Proposition 8. *Let Φ be the set of occurrences of x_j for some j in $\text{AI}_{\text{cl}}(C)$ for C in a resolution refutation. Then all terms at the positions Φ in the respective corresponding literal in C refer the same term.*

Proof. Proof by induction.

Base case: $\text{AI}_{\text{cl}}(C) = \ell[C]$ for C in some initial clause set. As x_j replaces a distinct term t_j , all occurrences of x_j in $\text{AI}_{\text{cl}}(C)$ correspond to some t_j in C .

Induction step: Suppose the statement holds for C_1 and C_2 , usual resolution step notation. Every $x_j \in C$ is derived from some $t \in C_1$ (w.l.o.g.).

- Suppose t is a variable replacing a colored term. Then $t = x_{j'}$ such that $\ell[x_{j'}\sigma]\tau = \ell[x_{j'}]\tau = x_{j'}\tau = x_j$.

Suppose that x_j refers to a colored term without free variables t_j . TODO □

Proposition 9. *Let C_1 and C_2 be clauses of a resolution step such that a literal $l \in C_1$ and $l' \in C_2$ are resolved upon using σ such that $l\sigma = l'\sigma$. Let $l_{\text{cl}} \in \text{AI}_{\text{cl}}(C_1)$ correspond to l and $l'_{\text{cl}} \in \text{AI}_{\text{cl}}(C_2)$ correspond to l' (cf. Lemma 11). Then $\text{au}(l_{\text{cl}}, l'_{\text{cl}})$ is well-defined, i.e. if it maps a variable x to another variable y , then y is unique.*

Proof. Suppose that $\{x_j \mapsto x_k\} \in \text{au}(l_{\text{cl}}, l'_{\text{cl}})$. Then either a (sub)term in l_{cl} or l'_{cl} is x_j and the corresponding (sub)term in the other literal is t such that $\ell[t\sigma] = x_k$, or ...

$\text{au}(l_{\text{cl}}, l'_{\text{cl}})$ is ill-defined if and only if there is another occurrence of x_j , at whose corresponding sub(term) in the other literal is s such that $\ell[s\sigma] = x_l$ with $k \neq l$. However as $l\sigma = l'\sigma$, we know that $t\sigma =$

BIG TODO □

Apparently, τ establishes equality for the terms in the literals being resolved on and quasi-equality for other literals in the remaining clause.

erals_clauses_equal)

Lemma 10. *Let l_{cl}, l'_{cl} be the literal in $AI_{cl}(C_1)$ and $AI_{cl}(C_2)$ corresponding to l and l' where C is the result of a resolution step from $C_1 : D \vee l$ and $C_2 : E \vee \neg l'$ to $C = (D \vee E)\sigma$ (i.e. $l\sigma = l'\sigma$). Let $\tau = \text{au}(l_{cl}\sigma, l'_{cl}\sigma)$. Then $\ell[l_{cl}\sigma]\tau = \ell[l'_{cl}\sigma]\tau$. // true and used*

Proof. Let s_{cl} be a (sub-)term of a parameter of $l_{cl}\sigma$ and $t_{cl}\sigma$ the term at the same term position in $l'_{cl}\sigma$. Let s and t be their corresponding (sub)-term at the same term position in $l\sigma$ and $l'\sigma$. We show that $\ell[s_{cl}\sigma]\tau = \ell[t_{cl}\sigma]\tau$ by induction on the structure of s_{cl} and t_{cl} respectively.

Note that by Lemma 3, s_{cl} and t_{cl} do not contain colored terms. Hence only grey terms can contain subterms.

Suppose that $s_{cl} = z_i$ and/or $t_{cl} = z_j$ for some i and j . Suppose that $s_{cl} \neq t_{cl}$ as otherwise we are done. Hence $s \neq t$, but by the resolution rule application $s\sigma = t\sigma$. Cases:

- $s_{cl} = z_i$ and $t_{cl} = z_j$ with $i \neq j$. As σ affects neither s_{cl} , nor t_{cl} , $\ell[s_{cl}\sigma] = s_{cl}$ and $\ell[t_{cl}\sigma] = t_{cl}$. We show that $s_{cl}\tau = t_{cl}\tau$.

Note that the function au visits all subterms and combines all mappings it encounters. Hence $\text{au}(s_{cl}, t_{cl})$ is part of the final substitution τ . However due to the just established circumstances, $\text{au}(s_{cl}, t_{cl}) = \{z_i \mapsto z_m, z_j \mapsto z_m\}$ with m as in the definition of au , so $s_{cl}\tau = t_{cl}\tau$.

- W.l.o.g. $s_{cl} = z_i$ and $t_{cl} \neq z_j$ for any j . Then as $s\sigma = t\sigma$ and s is a Δ -term, t must be a variable which is substituted for a Δ -term by σ . So $\ell[t\sigma] = z_k$ for some k . Note that the function au visits all subterms and combines all mappings it encounters. At $\text{au}(s_{cl}, t_{cl})$, $\{z_i \mapsto z_k\}$ is added. Therefore $\ell[s_{cl}\sigma]\tau = \ell[z_i\sigma]\tau = \ell[z_i]\tau = z_i\tau = z_k$. As the Δ -term which is replaced by z_k occurs in the unified literal, none of its occurrences are modified any further, so z_k is not updated to some z_l , $l \neq k$ by the substitution τ . Therefore $z_k = z_k\tau = \ell[t_{cl}\sigma]\tau = z_k\tau$.

Suppose that s_{cl} and t_{cl} are grey terms: Then their outermost symbol is unaffected by the lifting and as $s\sigma = t\sigma$, the outermost symbol is the same under σ in $s_{cl}\sigma$ and $t_{cl}\sigma$. The equality of the parameters is established by the induction hypothesis. Note that grey constants can be treated as grey functions without parameters.

Suppose that one of s_{cl} and t_{cl} are grey terms. W.l.o.g. Let s_{cl} be a grey term. Then as $s\sigma = t\sigma$, t_{cl} is a variable such that $t_{cl}\sigma$ is a grey term. Even more, their outermost symbol is the same due to $s\sigma = t\sigma$. Equality of the children is established by the induction hypothesis.

Suppose that both s_{cl} and t_{cl} are variables. Suppose that σ is non-trivial on at least s_{cl} or t_{cl} , as otherwise we would be done. Note that due to $s\sigma = t\sigma$, the outermost symbol of both $s_{cl}\sigma$ is the same as the one of $t_{cl}\sigma$. As the equality of the children of $s_{cl}\sigma$ and $t_{cl}\sigma$ is established by the induction hypothesis, we are done. \square

:literal_in_clause)

Lemma 11. *If a literal l occurs in a clause C from a resolution refutation, then $AI_{cl}(C)$ contains a literal l_{cl} such that $l_{cl} \sim \ell[l]$, where \sim means equal up to the index of the variables which replace colored terms. // true but unused*

Proof. Base case: By Definition of AI_{cl} .

Let C be the result of a resolution step from $C_1 : D \vee l$ and $C_2 : E \vee \neg l'$ to $C = (D \vee E)\sigma$. Every literal of C is derived from a literal in C_1 or C_2 . Let λ be a literal in C_1 . The case for a literal in C_2 is analogous. Note that $\lambda \neq l$ as otherwise it would not be contained in C .

By assumption $\lambda \in C_1$. Then by the resolution rule application, $\lambda\sigma \in C$.

By the induction hypothesis, $\ell[\lambda] \in AI_{cl}(C_1)$. By the definition of AI_{cl} , $\ell[\ell[\lambda]\sigma]\tau \in AI_{cl}(C)$ with $\tau = au(l, l')$.

So we have to show that $\ell[\lambda\sigma] \sim \ell[\ell[\lambda]\sigma]\tau$.

We perform an induction on the depth of terms in λ .

- Suppose t is a term of size 1 in λ and it is a variable.

Suppose the variable t was introduced by a lifting, say z_i . Then $t\sigma = t$ and $\ell[t\sigma] = t$. Furthermore, $\ell[t] = t$, $\ell[t]\sigma = t$ and $\ell[\ell[t]\sigma] = t$. As τ only replaces variables which replace colored terms by other variables of the same type, $\ell[\lambda\sigma] \sim \ell[\ell[\lambda]\sigma]\tau$. **NB:** this is where variable indices may not match.

Otherwise suppose the variable t occurs in the resolution refutation, say u . Suppose that σ is not trivial on u as otherwise we are done. Note that by the induction hypothesis, we get the desired relation for possible subterms of $u\sigma$. Suppose that $u\sigma$ is a grey term. Then $\ell[u\sigma] = u\sigma$.

Furthermore as $\ell[u] = u$, $\ell[u]\sigma = u\sigma$. Hence $\ell[u\sigma] = \ell[\ell[u]\sigma]$. As τ again only replaces variables which replace colored terms by other variables of the same kind, the induction holds in this case. **NB:** τ stuff again

- Suppose t is a term of size 1 in λ and it is a constant. Suppose t is grey. Then it is unaffected by both liftings and substitutions. Otherwise suppose t is colored. Then $\ell[t\sigma] = \ell[t] = \ell[t]\sigma = \ell[\ell[t]\sigma]$. As τ again only replaces variables which replace colored terms by other variables of the same kind, the induction holds in this case. **NB:** τ stuff again

- Suppose t is of the form $f(t_1, \dots, t_n)$. Then by the induction hypothesis, $\ell[t_i\sigma] = \ell[\ell[t_i]\sigma]\tau$ for $1 \leq i \leq n$.
 - Suppose f is grey. Then f is neither affected by substitutions nor by liftings.
 - Suppose f is colored. We only consider the case of occurrences of maximal colored terms as the other ones are discarded by the lifting. Then $\ell[t\sigma] = z_i$ for some i and $\ell[t] = z_j$ for some j . Furthermore $\ell[t]\sigma = z_j$ and $\ell[\ell[t]\sigma] = z_j$. Hence $\ell[t\sigma] \sim \ell[\ell[t]\sigma]$. \square

Lemma 12. *Suppose no Γ -term occurs in Γ . Then $\Gamma \models \text{AI}_{\text{mat}}(C) \vee \text{AI}_{\text{cl}}(C)$.*

Proof. Proof by induction of the strenghtening: $\Gamma \models \text{AI}_{\text{mat}}(C) \vee \text{AI}_{\text{cl}}(C_\Gamma)$.

Base case:

For $C \in \Gamma$, $\text{AI}_{\text{mat}}(C) = \perp$ and $\text{AI}_{\text{cl}}(C) = \ell[C_\Gamma] = \ell_{\Gamma,y}[C]$. By the restriction, $\ell_{\Gamma,y}[C] = C$ and $\Gamma \models C$.

For $C \in \Delta$, $\text{AI}_{\text{mat}}(C) = \top$.

Induction step:

Resolution.

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

We introduce the following abbreviations, where l^* is the literal in AI_{cl} which corresponds to l :

$$\text{AI}_{\text{cl}}(C_1)^* = \text{AI}_{\text{cl}}((C_1)_\Gamma) \setminus \{\ell[(l_{\text{cl}})_\Gamma]\}$$

$$\text{AI}_{\text{cl}}(C_2)^* = \text{AI}_{\text{cl}}((C_2)_\Gamma) \setminus \{\ell[\neg(l'_{\text{cl}})_\Gamma]\}$$

$$\tau = \text{au}((l_{\text{cl}})_\Gamma, (l'_{\text{cl}})_\Gamma)$$

$$\text{AI}_{\text{cl}}(C_\Gamma) = \ell[(\text{AI}_{\text{cl}}(C_1)^* \vee \text{AI}_{\text{cl}}(C_2)^*)\sigma]\tau.$$

$$\text{By Lemma 2, } \text{AI}_{\text{cl}}(C_\Gamma) = \ell[\text{AI}_{\text{cl}}(C_1)^*\sigma]\tau \vee \ell[\text{AI}_{\text{cl}}(C_2)^*\sigma]\tau.$$

By the induction hypothesis, $\Gamma \models \text{AI}_{\text{mat}}(C_i) \vee \text{AI}_{\text{cl}}(C_{i\Gamma})$, $i \in \{1, 2\}$, or expressed differently:

$$\Gamma \models \text{AI}_{\text{mat}}(C_1) \vee \text{AI}_{\text{cl}}(C_1)^* \vee (l_{\text{cl}})_\Gamma$$

$$\Gamma \models \text{AI}_{\text{mat}}(C_2) \vee \text{AI}_{\text{cl}}(C_2)^* \vee \neg(l'_{\text{cl}})_\Gamma$$

By Lemma 3, $\text{AI}_{\text{mat}}(C_1)$ and $\text{AI}_{\text{cl}}(C_1)$ as well as $\text{AI}_{\text{mat}}(C_2)$ and $\text{AI}_{\text{cl}}(C_2)$ do not contain colored terms. Hence by Lemma 6, Lemma 2 and applying τ , we get that

$$\Gamma \stackrel{(\circ)}{\models} \ell[\text{AI}_{\text{mat}}(C_1)\sigma]\tau \vee \ell[\text{AI}_{\text{cl}}(C_1)^*\sigma]\tau \vee \ell[(l_{\text{cl}})_\Gamma\sigma]\tau$$

restriction
used here

$$\Gamma \models^{(*)} \ell[\text{AI}_{\text{mat}}(C_2)\sigma]\tau \vee \ell[\text{AI}_{\text{cl}}(C_2)^*\sigma]\tau \vee \neg\ell[(l'_{\text{cl}})_{\Gamma}\sigma]\tau$$

By Lemma 10, $\ell[(l_{\text{cl}})_{\Gamma}\sigma]\tau = \ell[(l'_{\text{cl}})_{\Gamma}\sigma]\tau$.

- If l and l' grey:

$$\text{AI}_{\text{mat}}(C) = (\neg\ell[l_{\text{cl}}\sigma]\tau \wedge \ell[\text{AI}_{\text{mat}}(C_1)\sigma]\tau) \vee (\ell[l_{\text{cl}}\sigma]\tau \wedge \ell[\text{AI}_{\text{mat}}(C_2)\sigma]\tau)$$

Suppose for a model M of Γ that $M \not\models \text{AI}_{\text{cl}}(C)$, i.e. $M \not\models \ell[\text{AI}_{\text{cl}}(C_1)\sigma]\tau$ and $M \not\models \ell[\text{AI}_{\text{cl}}(C_2)\sigma]\tau$ as otherwise we would be done. Then by (\circ) and $(*)$:

$$M \models \ell[\text{AI}_{\text{mat}}(C_1)\sigma]\tau \vee \ell[l_{\text{cl}}\sigma]\tau$$

$$M \models \ell[\text{AI}_{\text{mat}}(C_2)\sigma]\tau \vee \neg\ell[l'_{\text{cl}}\sigma]\tau$$

By Lemma 10, $\ell[l_{\text{cl}}\sigma]\tau = \ell[l'_{\text{cl}}\sigma]\tau$. By a case distinction on the truth value of $\ell[l_{\text{cl}}\sigma]\tau$ in M , we obtain that $M \models \text{AI}_{\text{mat}}(C)$.

- If l and l' are Γ -colored: $\text{AI}_{\text{mat}}(C) = \ell[(\text{AI}_{\text{mat}}(C_1) \vee \text{AI}_{\text{mat}}(C_2))\sigma]\tau$

By Lemma 10, we can do a resolution step on $\ell[l_{\text{cl}}\sigma]\tau$ of (\circ) and $(*)$ to arrive at

$$\Gamma \models \ell[\text{AI}_{\text{mat}}(C_1)\sigma]\tau \vee \ell[\text{AI}_{\text{cl}}(C_1)^*\sigma]\tau \vee \ell[\text{AI}_{\text{mat}}(C_2)\sigma]\tau \vee \ell[\text{AI}_{\text{cl}}(C_2)^*\sigma]\tau$$

This however is by Lemma 2 nothing else than $\Gamma \models \text{AI}_{\text{mat}}(C) \vee \text{AI}_{\text{cl}}(C)$

- If l and l' are Δ -colored: $\text{AI}_{\text{mat}}(C) = \ell[(\text{AI}_{\text{mat}}(C_1) \wedge \text{AI}_{\text{mat}}(C_2))\sigma]\tau$

As l is Δ -colored, (\circ) and $(*)$ reduce to:

$$\Gamma \models \ell[\text{AI}_{\text{mat}}(C_1)\sigma]\tau \vee \ell[\text{AI}_{\text{cl}}(C_1)^*\sigma]\tau$$

$$\Gamma \models \ell[\text{AI}_{\text{mat}}(C_2)\sigma]\tau \vee \ell[\text{AI}_{\text{cl}}(C_2)^*\sigma]\tau$$

But this implies that

$$\Gamma \models (\ell[\text{AI}_{\text{mat}}(C_1)\sigma]\tau \wedge \ell[\text{AI}_{\text{cl}}(C_1)^*\sigma]\tau) \vee \ell[\text{AI}_{\text{cl}}(C_1)^*\sigma]\tau \vee \ell[\text{AI}_{\text{cl}}(C_2)^*\sigma]\tau$$

This however is by Lemma 2 nothing else than $\Gamma \models \text{AI}_{\text{mat}}(C) \vee \text{AI}_{\text{cl}}(C)$. \square

3 thoughts

Conjecture 13. *Double ended arrows are not important as terms are overbound with same variable anyway as always same unifier applies.*

4 arrow proof

for_same_variables)

Lemma 14. *Whenever the same variable appears multiple times in $\text{PI}(C) \vee C$ for $C \in \pi$, there are arrows.*

- *If both variables are contained only in grey terms, there is a double arrow // they unify to exactly the same*
- *If only one variable is only contained in grey terms, there is an arrow from it to the other one // either unify the one in grey term, then other one must be overbound later. if otherwise var in the colored term is unified, we can still overbind the grey one first.*
- *otherwise there are weak arrows between them // have same quantifier, so order does not matter, but want to keep dependencies on both the same*

Proof. By induction. Note: As required by resolution, all initial clauses are variable disjoint.

Base case: In the initial clause sets, consider for a clause C two different positions p_1 and p_2 pointing to the same variable. Then either:

- p_1 and p_2 contain only grey symbols. Then $(p_1, p_2) \in \mathcal{A}$.
- Only p_i , $i \in \{1, 2\}$ contains only grey symbols. Then $(p_i, p_{(i \bmod 2)+1}) \in \mathcal{A}$.
- There are not only grey symbols in both p_1 and p_2 , i.e. both contain at least a colored symbol. Then $\{p_1, p_2\} \in \mathcal{W}$.

Induction step: Suppose a clause C is the result of a resolution of $C_1 : D \vee l$ and $C_2 : E \vee \neg l$ with $l\sigma = l'\sigma$. $\text{PI}(C)$ is $[\text{PI}(C_1) \circ \text{PI}(C)]\sigma$ or $[(l \wedge \text{PI}(C_2)) \vee (\neg l \wedge \text{PI}(C_1))]\sigma$.

Assumption: C_1 and C_2 are variable disjoint, i.e. variables are renamed in case C_1 and C_2 are derived from some common original clause and share variables.

By the induction hypothesis, there are appropriate arrows in both $\text{PI}(C_i) \vee C_i$, $i \in \{1, 2\}$. If the variables were present in C_1 or C_2 , the arrow is still there, either in $\text{PI}(C)$ (in the case of l or l'), C (in case of D and E) or in currently not shown literal (in case l and l' have the same color).

Otherwise, it was introduced by unification in $l\sigma$ or $\text{PI}(C_i)\sigma$. In this case, there is some term position q in with $\rho(l).q$ a variable and $\rho(l').q$ a variable or a term containing variables (or other way around). Hence unification maps a variable to a variable or a term containing variables. The variable being unified is in $\text{PI}(C_i) \vee C_i$ for some $i \in \{1, 2\}$.

But by the induction hypothesis, all occurrences of each variable does already have appropriate arrows, which are still present. \square

Lemma 15. *In $\text{PI}(C) \vee C$ for $C \in \pi$, if there is a Δ -colored term s in a Γ -term t , then there is an arrow from p_1 to p_2 such that $\rho(p_1) = s$ and $\rho(p_2) = s$ and for some i , $\rho(p_2 \bmod i) = t$.*

Note: p_1 might be in some clause, the prop interpolant or none of both.

Proof. By induction.

Base case: There are no foreign terms in the initial clause sets, so no arrows necessary.

Induction step:

Resolution. Suppose a clause C is the result of a resolution of $C_1 : D \vee l$ and $C_2 : E \vee \neg l$ with $l\sigma = l'\sigma$.

1. Suppose l is colored. This case is similar to the grey one, with the exception that the cases applying to l in PI do not apply.
2. Suppose l is grey. Then $\text{PI}(C) = [(l \wedge \text{PI}(C_2)) \vee (\neg l \wedge \text{PI}(C_1))]\sigma$

By the induction hypothesis, there are appropriate arrows in $\text{PI}(C_1) \vee C_1$ and $\text{PI}(C_2) \vee C_2$.

We show that for all maximal Γ -terms in $\text{PI}(C) \vee C$ with Δ -terms in them which were not present in $\text{PI}(C_i) \vee C_i$, $i \in \{1, 2\}$, there is an arrow.

Γ -terms and Δ -terms are not unifiable. Hence all pairs of terms (t_1, t_2) in the same positions in l and l' (if both positions exist) either point to the same symbol or (w.l.o.g.) t_1 is a variable and t_2 is a term. If there are Δ -terms in Γ -terms in the prefix, they are present in both ancestors and handled by the induction hypothesis.

The only way a Δ -colored term may enter a Γ -colored term is in the situation where t_1 is a variable and t_2 a colored term. But then $\text{mgu}(t_1, t_2)$ applied to t_1 yields t_2 , i.e. “the parts of σ concerned with unifying t_1 and t_2 ” do not introduce new Δ -terms in Γ -terms. In other words, all such situation have been present in $\text{PI}(C_i) \vee C_i$ for $i \in \{1, 2\}$ and since the arrows for l and l' are merged, they are present for $l\sigma$ in $\text{PI}(C)$.

This handles the case where terms t_1 and t_2 are unified. But unification also affects all other occurrences of variables, this means “the parts of σ not concerned with unifying t_1 and t_2 ”. The relevant case for this lemma is when a

Γ -term contains a variable, that is substituted by a term containing Δ -terms. But in this case, by Lemma 14, there is an arrow from the other occurrence of the variable to the one in the Γ -term: either double arrow in \mathcal{A} if both prefixes are grey, one in \mathcal{A} if one of the prefixes is grey or one in \mathcal{W} if both prefixes contain a colored symbol. \square