

TD 6 – Calcul des séquents, Unification

Nicolas Dumange nicolas.dumange@ens-paris-saclay.fr
Théo Vignon theo.vignon@ens-paris-saclay.fr

Nous considérons le calcul des séquents tel que vu dans le cours, c'est-à-dire le calcul des séquents sans coupures.

Exercice 1 — Preuves en calcul des séquents

Donnez des preuves dans le calcul des séquents des formules suivantes:

1. $a \vee (a \Rightarrow b)$
2. $a \vee (b \wedge c) \Leftrightarrow (a \vee b) \wedge (a \vee c)$
3. $((a \Rightarrow b) \Rightarrow a) \Rightarrow a$
4. $\neg\neg a \Rightarrow a$
5. $\neg(a \vee b) \Leftrightarrow \neg a \wedge \neg b$
6. $\neg\forall x, r(x) \Leftrightarrow \exists x, \neg r(x)$
7. $\forall x, (q \vee r(x)) \Rightarrow (q \vee \forall x, r(x))$
8. $\exists x, [(r(a) \vee r(b)) \Rightarrow r(x)]$
9. Montrez que si $a \neq b$, alors il n'existe pas de preuve de 8. qui n'utilise pas de contractions.

Exercice 2 — Quelques exemples d'unification

Définition — Problème d'unification

Un *problème d'unification* est un ensemble E d'équations de la forme $t \stackrel{?}{=} u$.

Un *unificateur* (i.e. une solution au problème d'unification) d'un ensemble E est une substitution σ telle que pour toute équation $t \stackrel{?}{=} u$ dans E , $t\sigma = u\sigma$.

La procédure d'unification vue en classe est rappelée à la fin de la feuille de td.

Appliquez la procédure aux problèmes d'unification suivants (votre réponse devrait être soit **fail** soit la substitution retournée par la procédure) (On considère que g, f, h, a sont des symboles de fonctions):

1. $E_1 = \left\{ f(x, g(a, y)) \stackrel{?}{=} f(h(y), g(y, a)), g(x, h(y)) \stackrel{?}{=} g(z, z) \right\}$
2. $E_2 = \left\{ f(x, x) \stackrel{?}{=} f(g(y), z), h(z) \stackrel{?}{=} h(y) \right\}$
3. $E_3 = \left\{ f(x, a) \stackrel{?}{=} f(b, y), f(x) \stackrel{?}{=} f(y) \right\}$

Exercice 3 — Étude de l'algorithme d'unification

Nous étudions les propriétés de l'algorithme d'unification.

1. L'algorithme présenté est-il déterministe?
2. Montrez que l'algorithme termine (en échouant ou en renvoyant une substitution).

On cherche maintenant à prouver que l'algorithme calcule effectivement un unificateur de l'ensemble d'équations donné en entrée (quand ceci existe). nous prouvons même un résultat plus général.

3. Montrez que si e contient $f(u_1, \dots, u_n) \stackrel{?}{=} g(t_1, \dots, t_m)$, avec $f \neq g$, alors e n'est pas unifiable.
4. Montrez que si e contient $x \stackrel{?}{=} t$, où $x \in \mathcal{X}$, $t \notin \mathcal{X}$ et $x \in \text{var}(t)$, alors e n'est pas unifiable.

Définition — mgu

un *unificateur le plus général* (mgu) σ d'un problème d'unification e est un unificateur de e tel que pour tout unificateur τ de e , il existe η telle que $\tau = \eta \circ \sigma$.

5. Montrez que le problème d'unification $x \stackrel{?}{=} f(y)$ possède une infinité d'unificateurs. En existe-t-il un le plus général? Lorsqu'il existe, y a-t-il unicité du mgu?
6. Montrez que si $e = e' \cup \{x \stackrel{?}{=} t\}$, alors σ unifie e ssi σ unifie e' . Observez de plus que ceci implique que σ est un mgu de e ssi σ est un mgu de e' .
7. Montrez que si $e = e' \cup \{f(u_1, \dots, u_n) \stackrel{?}{=} f(t_1, \dots, t_n)\}$, alors σ unifie e ssi σ unifie $e' \cup \{u_1 \stackrel{?}{=} t_1, \dots, u_n \stackrel{?}{=} t_n\}$. Observez de plus que ceci implique que σ est un mgu de e ssi σ est un mgu de e' .
8. Montrez que pour toute substitution σ , variable x et terme t , si $x\sigma = t\sigma$ alors $\sigma = \sigma \circ \{x \mapsto t\}$.
9. Montrez que si $e = e' \cup \{x \stackrel{?}{=} t\}$ où $x \notin \text{var}(t)$, alors σ unifie e' ssi $\sigma \circ \{x \mapsto t\}$ unifie e .
10. Montrez grâce à la question précédente que: σ est un mgu de e' implique que $\sigma \circ \{x \mapsto t\}$ est un mgu de e .
11. Montrez, étant donné une entrée e , que
 - si l'algorithme retourne une substitution σ , alors σ est un mgu de e
 - si l'algorithme échoue, alors le problème d'unification e n'a pas de solution.

Exercice 4 — Théorème d'interpolation

Si φ est une formule, on appelle $l(\varphi)$ l'ensemble des variables libres, symboles de fonction et symboles de prédicat apparaissant dans φ . Par extension, si γ est un multi-ensemble de formules, on définit $l(\gamma) = \bigcup_{\varphi \in \gamma} l(\varphi)$.

On souhaite montrer que si $l(\gamma_1 \cup \gamma_2 \cup \delta_1 \cup \delta_2)$ ne contient pas de symboles de fonction, et si $\gamma_1, \gamma_2 \vdash \delta_1, \delta_2$ est prouvable, alors il existe une formule ξ telle que:

- $\gamma_1 \vdash \xi, \delta_1$ and $\gamma_2, \xi \vdash \delta_2$ sont prouvables
- $l(\xi) \subseteq l(\gamma_1 \cup \delta_1) \cap l(\gamma_2 \cup \delta_2)$

1. Prouvez ce résultat. On considèrera en particulier les cas suivants: ax ; $\Rightarrow_{\text{right}}$; $\Rightarrow_{\text{left}}$; $\forall_{\text{left}}$; $\forall_{\text{right}}$; $\exists_{\text{left}}$; $\exists_{\text{right}}$.

2. Prouvez le *théorème d'interpolation*: si φ et ψ sont des formules sans symboles de fonctions, et si $\vdash \varphi \Rightarrow \psi$ est prouvable, alors il existe une formule ξ telle que:

- $\vdash \varphi \Rightarrow \xi$ et $\vdash \xi \Rightarrow \psi$ sont prouvables
- $l(\xi) \subseteq l(\varphi) \cap l(\psi)$.

3. (*) Nous appliquons ceci à la preuve du théorème de *Beth*.

Soient p et p' , soit $\gamma(p)$ un ensemble de formules closes ne contenant pas le symbole p' ou de symboles de fonctions. On note $\gamma(p')$ l'ensemble de symboles obtenu en remplaçant toutes les occurrences de p par p' dans $\gamma(p)$.

On dit que $\gamma(p)$ *définit implicitement* p si $\gamma(p), \gamma(p') \vdash \forall x. (p(x) \Leftrightarrow p'(x))$ est prouvable. On dit que $\gamma(p)$ *définit explicitement* p s'il existe une formule $\varphi(x)$ n'utilisant ni p ni p' telle que $\gamma(p) \vdash \forall x. (\varphi(x) \Leftrightarrow p(x))$.

Prouvez que $\gamma(p)$ définit implicitement p si et seulement si $\gamma(p)$ définit explicitement p .

4. (**) En réalité le théorème d'interpolation reste vrai même quand les formules impliquées contiennent des symboles de fonctions. Voyez-vous comment adapter la preuve à ce cas ?

Définition — Unification

Unif(E : Unification problem):

```

1 if  $E = E' \cup \{f(u_1, \dots, u_n) \stackrel{?}{=} f(t_1, \dots, t_n)\}$  then
2 |   return Unif( $E' \cup \{u_1 \stackrel{?}{=} t_1, \dots, u_n \stackrel{?}{=} t_n\}$ )
3 else if  $E = E' \cup \{f(u_1, \dots, u_n) \stackrel{?}{=} g(t_1, \dots, t_m)\}$  (where  $f \neq g$ ) then
4 |   return fail
5 else if  $E = E' \cup \{x \stackrel{?}{=} x\}$  then
6 |   return Unif( $E'$ )
7 else if  $E = E' \cup \{x \stackrel{?}{=} t\}$  where  $x \in \mathcal{X}$  and  $x \notin \text{Var}(t)$  then
8 |   return Unif( $E' \cup \{x \mapsto t\}$ )  $\cup \{x \mapsto t\}$ 
9 else if  $E = E' \cup \{x \stackrel{?}{=} t\}$  where  $x \in \mathcal{X}, t \notin \mathcal{X}$  and  $x \in \text{Var}(t)$  then
10 |  return fail
11 else if  $E = \emptyset$  then
12 |  return id
```