Algorithme d'unification

Manon Ruffini

Définition 1

Deux termes u et v sont unifiables lorsque : il existe une substitution σ telle que :

$$u[\sigma] = v[\sigma]$$

Théorème 1

Soient u et v deux termes unifiables. Alors, il existe un unificateur σ tel que pour tout autre unificateur σ' de u et v, il existe une substitution σ'' telle que :

$$\sigma' = \sigma'' \circ \sigma$$

*Un tel unificateur est appelé **unificateur principal**.*

On veut un algorithme qui calcule cet unificateur principal.

Étape 1

Présentation de l'algorithme

Algorithme 1 : Unification

Entrées : Un ensemble d'équations $E = \{u_1 \sim v_1, \ldots, u_n \sim v_n\}$, où les u_i, v_i sont des termes
Résultat : L'unificateur principal σ de E, s'il existe ; Échec sinon

```
σ ← Id;
tant que E ≠ ∅ faire
    Choisir e ∈ E;
    E' ← E \ {e};
    /* Cas 1                                              */
    si e = f(u_1 ... u_p) ~ g(v_1 ... v_q) alors
        si f = g alors
            E ← E' ∪ {u_1 ~ v_1, ..., u_p ~ v_p}
        sinon retourner "Échec 1";
    /* Cas 2                                              */
    si e = x ~ x alors
        E ← E'
    /* Cas 3                                              */
    si e = x ~ u ou e = u ~ x alors
        si x n'apparaît pas dans u alors
            σ ← [x := u] ∘ σ;
            E ← E'[x := u]
        sinon retourner "Échec 2";
fin
retourner σ
```

Dans la suite on notera : E_n et σ_n les valeurs respectives de E et σ après le n-ème tour de boucle, pour $n \geqslant 1$; et E_0 et σ_0 pour leur valeur initiale.

Étape 2

Cet algorithme termine.

Pour tout $n \in \mathbb{N}$, soient :

— a_n le nombre de variables distinctes de E_n
— b_n le nombre de symboles de fonctions de E_n
— c_n le nombre d'équations de E_n

Soit $n \in \mathbb{N}$. Supposons que E_n et E_{n+1} soient bien définis. Alors :

— Si on est dans le cas 1, on a : $a_{n+1} = a_n$ et $b_{n+1} < b_n$
— Si on est dans le cas 2, on a : $a_{n+1} \leqslant a_n$, $b_{n+1} = b_n$ et $c_{n+1} < c_n$
— Si on est dans le cas 3, on a : $a_{n+1} < a_n$

Ainsi,

$$(a_{n+1}, b_{n+1}, c_{n+1}) \prec (a_n, b_n, c_n)$$

où $\prec$ désigne l'ordre lexicographique strict sur $\mathbb{N}^3$.
Comme cet ordre est bien fondé, l'algorithme termine.

Étape 3

Si $x[\sigma] = u[\sigma]$, alors : $\sigma = \sigma \circ [x := u]$.

Soit $\sigma' = \sigma \circ [x := u]$. Alors : $x[\sigma'] = x[x := u][\sigma] = u[\sigma] = x[\sigma]$ Et pour toute variable $y \neq x$, $y[\sigma'] = y[x := u][\sigma] = y[\sigma]$ Donc

$$\sigma = \sigma'$$

Étape 4

Pour tout $n \in \mathbb{N}$, on définit la propriété suivante :
$(H_n) :=$ "Si E_n et σ_n sont bien définis, alors : σ unifie E ssi il existe un unificateur σ' de E_n tel que $\sigma = \sigma' \circ \sigma_n$"

On fait une récurrence sur n :

$n = 0$: $\sigma_0 = \mathrm{Id}$ et $E_0 = E$, donc H_0 est immédiat

Soit $n \in \mathbb{N}$: Supposons H_n et que E_{n+1} et σ_{n+1} existent. Alors, par hypothèse :

$$\sigma \text{ unifie } E \Leftrightarrow \exists \sigma', \sigma = \sigma' \circ \sigma_n \text{ et } \sigma' \text{ unifie } E_n$$

On veut montrer

$$\sigma \text{ unifie } E \Leftrightarrow \exists \sigma'', \sigma = \sigma'' \circ \sigma_{n+1} \text{ et } \sigma'' \text{ unifie } E_{n+1}$$

Donc montrons :

$$\exists \sigma', \sigma = \sigma' \circ \sigma_n \text{ et } \sigma' \text{ unifie } E_n \Leftrightarrow \exists \sigma'', \sigma = \sigma'' \circ \sigma_{n+1} \text{ et } \sigma'' \text{ unifie } E_{n+1}$$

Cas 1 : $E_n = E' \uplus \{f(u_1, \ldots, u_p), f(v_1, \ldots, v_p)\}$, et $E_{n+1} = E' \cup \{u_1 \sim v_1, \ldots, u_p \sim v_p\}$, et $\sigma_{n+1} = \sigma_n$. On a :
Par définition (inductive) de substitution sur les termes : σ' unifie E_n ssi σ' unifie E_{n+1}.
De plus $\sigma = \sigma' \circ \sigma_n = \sigma' \circ \sigma_{n+1}$ car $\sigma_{n+1} = \sigma_n$.

Cas 2 : $E_{n+1} = E'$, et $\sigma_{n+1} = \sigma_n$. Le résultat est immédiat.

Cas 3 : On a : $E_n = E' \uplus \{x \sim u\}$ ou $E' \uplus \{u \sim x\}$, avec $x \neq u$ et x n'apparaît pas dans u.

" $\Leftarrow$ " Il existe σ'' telle que $\sigma = \sigma'' \circ \sigma_{n+1}$ et σ'' unifie E_{n+1}. Alors $\sigma = \sigma'' \circ [x := u] \circ \sigma_n$; et $\sigma'' \circ [x := u]$ unifie E_n donc $\sigma' = \sigma'' \circ [x := u]$ convient.

" $\Rightarrow$ " Soit σ' qui unifie E_n. En particulier, $x[\sigma'] = u[\sigma']$. D'après l'étape 3, $\sigma' = \sigma' \circ [x := u]$. Donc, $\sigma' = \sigma' \circ \sigma_n = \sigma' \circ [x := u] \circ \sigma_n = \sigma' \circ \sigma_{n+1}$. Donc $\sigma'' = \sigma'$ convient.

On a bien montré H_n, pour tout n.

Étape 5

L'algorithme est correct.

On suppose que l'algorithme effectue N boucles avant de terminer.

— Si $E_N = \emptyset$ (pas d'échec) : Id unifie E_N et d'après H_N, comme $\sigma_N = \mathrm{Id} \circ \sigma_N$, σ_N unifie E. De plus, si σ est un unificateur de E, d'après H_N, il existe une substitution σ' telle que $\sigma = \sigma' \circ \sigma_N$. Donc σ_N est bien un unificateur principal de E.

— Si l'algorithme renvoie "Échec 1", il existe un n tel que E_n contient $f(\dots) \sim g(\dots)$ avec $f \neq g$. Puisqu'une substitution ne change jamais les symboles de fonctions, E_n n'a pas d'unificateur. D'après H_n, E n'a pas d'unificateur.

— Si l'algorithme renvoie "Échec 2", il existe un n tel que E_n contient $x \sim u$, avec x dans u et $x \neq u$. Ainsi, comme u contient au moins un symbole de fonction, pour toute substitution σ, $|x[\sigma]| < |u[\sigma]|$, et donc $x[\sigma] \neq u[\sigma]$. Par H_n, E n'a pas d'unificateur.

Remarques

Ce développement est TRES long, et pour avoir le temps de bien faire la preuve, je pense que c'est raisonnable d'écrire l'algorithme dans le plan, et de seulement le rappeler à l'oral au début du développement. On peut aussi juste énoncer le résultat de l'étape 3 et détailler la preuve si on nous le demande.

Références

[1] R. David, K. Nour, C. Raffalli, *Introduction à la logique*. Dunod, 2è édition, 2004.