TD N° 4 : UNIFICATION

Exercice 1 - Termes et substitutions

Appliquer les substitutions suivantes

1. $\sigma_1 = [x \leftarrow 2]$
2. $\sigma_2 = [x \leftarrow 3; y \leftarrow c]$
3. $\sigma_3 = [x \leftarrow (g\ y); y \leftarrow 0; z \leftarrow h\ x\ x]$

aux termes ci-dessous

a. $f\ x\ 3$

b. $f\ x\ (g\ y)$

c. $f\ (h\ x\ y)\ (h\ x\ z)$.

Activité 2 - Unificateur le plus général

Considérons les termes $t_1 = g\ x\ (f\ y)$ et $t_2 = g\ x\ (f\ (f\ z))$ et les substitutions $\sigma_1 = [y \leftarrow f\ 4;\ z \leftarrow 4]$, $\sigma_2 = [y \leftarrow f\ z;\ x \leftarrow 7]$ et $\sigma_3 = [y \leftarrow f\ z]$. Nous nous intéressons au problème d'unification $t_1 \stackrel{?}{=} t_2$.

1. Déterminer $\sigma_1(t_1)$ et $\sigma_1(t_2)$, qu'observe-t-on ?
2. Même question pour σ_2 et σ_3.
3. Quelle substitution est candidate pour être le *mgu* de t_1 et t_2 ?

Exercice 3 - Unification

Nous considérons les termes suivants :

1. $f\ (g\ 3\ x)\ 2$
2. $f\ (g\ 3\ x)\ y$
3. $f\ x\ (g\ 3\ x)$
4. $f\ y\ (g\ 3\ z)$

Comme dans le cours, f, g sont des symboles de constantes, x, y, z sont des symboles de variables. Parmi ces termes, lesquels sont unifiables ? Déterminer leur *mgu*.

Exercice 4 - Unification

Faites l'unification des termes suivants :

1. $g\ x \stackrel{?}{=} g\ y$
2. $h\ y\ (g\ x) \stackrel{?}{=} h\ x\ y$
3. $g\ x\ (f\ a) \stackrel{?}{=} g\ y\ (f\ x)$
4. $g\ (f\ (f\ x))\ (f\ x) \stackrel{?}{=} g\ (f\ y)\ (f\ x)$

Comme dans le cours, a, f, g, h sont des symboles de constantes, x, y sont des symboles de variables.

Exercice 5 - extrait du CC 2017

Les questions 1 et 2 sont indépendantes.

1. Prouver la terminaison de l'algorithme d'Unification.
   On pourra considérer le nombre de variables de E et le multi-ensembles des tailles des égalités de E.
2. Les termes suivants sont-ils unifiables ? Indiquer précisément les règles utilisées à chaque étape.

$$h\ x\ (g\ a) \stackrel{?}{=} h\ y\ (g\ x), \quad h\ x\ (g\ y) \stackrel{?}{=} h\ y\ x, \quad h\ (g\ (g\ x))\ (g\ x) \stackrel{?}{=} h\ (g\ y)\ (g\ a)$$

   Rappelons que les lettres x, y désignent les variables, toutes les autres sont des constantes.

Activité : Preuves de l'algorithme d'Unification

Dans cette activité, on note (E_n, S_n) le résultat de l'algorithme à l'étape n.

Terminaison de l'algorithme

L'objectif est de démontrer le théorème suivant :

Théorème 1

L'algorithme d'Unification termine toujours, soit par un échec, soit avec $E_n = \emptyset$.

Nous l'avons vu, pour prouver la terminaison d'un algorithme, il faut trouver un variant appartenant à un ensemble muni d'un ordre bien fondé.

1. Une première idée est d'observer a_n le nombre d'équations de E_n, $a_n \in \mathbb{N}$ qui est muni d'un ordre bien fondé. Mais est-ce une bonne idée ?
2. Notons b_n le nombre de variables libres dans E_n, là encore, $b_n \in \mathbb{N}$ mais est-ce que ça constitue un bon variant ?
3. (a) Nous allons définir la taille d'un terme de la façon suivante, si x est une variable $|x| = 1$, si c est une constante, $|c| = 1$, et si t_1 et t_2 sont deux termes, alors $|(t_1\ t_2)| = |t_1| + |t_2|$. Quelle est la taille des termes suivants ?
      - $t_1 = f\ (g\ 3)\ y$
      - $t_2 = f\ x\ 2$

   (b) Soit $s \stackrel{?}{=} t$ un problème d'unification. On définit la taille de $s \stackrel{?}{=} t$ par $max(|s|, |t|)$. Quelle est la taille des problèmes d'unification suivants ?
      - $2 \stackrel{?}{=} x$
      - $g\ 3 \stackrel{?}{=} y$
      - $t_1 \stackrel{?}{=} t_2$

   (c) Rappelons que E_n est un multi-ensemble d'équations à unifier. Nous lui associons le multi-ensemble m_n des tailles de ces égalités. Par exemple, si $E_n = \{2 \stackrel{?}{=} x,\ g\ 3 \stackrel{?}{=} y,\ t_1 \stackrel{?}{=} t_2\}$, quel multi-ensemble lui associe-t-on ?

   (d) Nous avons vu en cours, un ordre bien fondé sur les multi-ensembles d'entiers naturels. En considérant le multi-ensemble formé des tailles des équations de E_n, a-t-on un bon variant ?
4. Pour construire un bon variant, nous allons composer les deux précédents candidats. À chaque étape de l'algorithme, nous associons à E_n un couple $(b_n, m_n) \in \mathbb{N} \times \mathscr{M}$ muni de l'ordre lexicographique qui est bien fondé. Démontrer alors le théorème.

Pour les plus rapides, Correction et Complétude

Soient deux termes s, t, les deux questions que l'on doit se poser après avoir prouver la terminaison sont les suivantes :

- Correction : Est-ce que l'algorithme fournit une substitution σ telle que $\sigma(s) = \sigma(t)$?
- Complétude : Est-ce qu'il fournit un unificateur si s et t sont unifiables ?

Ce qui se résume avec le théorème suivant :

Théorème 2

Pour deux termes s et t, l'algorithme d'unification est correct et complet, il calcule le *mgu* de s et t.

Une idée de la preuve.
On raisonne par récurrence sur la longueur de la dérivation, en utilisant le lemme suivant :

Lemme 1

· Si $(E,S) \Rightarrow (E',S')$, alors le *mgu* de (E,S) est égal au *mgu* de (E',S').
· Si $(E,S) \Rightarrow fail$, alors (E,S) n'a pas d'unificateurs.

Il faut prouver ce lemme pour chaque règle.

□

Règles de substitution

- **Delete :** $(\{t \stackrel{?}{=} t\} \cup E, S) \Rightarrow (E,S)$
- **Decompose :** $((s_1\ s_2) \stackrel{?}{=} (t_1\ t_2) \cup E, S) \Rightarrow (\{s_1 \stackrel{?}{=} t_1, s_2 \stackrel{?}{=} t_2\} \cup E, S)$
- **Fail :** $((s_1\ s_2) \stackrel{?}{=} c \cup E, S) \Rightarrow fail$
- **Clash :** $(c_1 \stackrel{?}{=} c_2 \cup E, S) \Rightarrow fail$
  si c_1 et c_2 sont des constantes différentes.
- **Eliminate :** $(x \stackrel{?}{=} t \cup E, S) \Rightarrow (E[x \leftarrow t], S[x \leftarrow t] \cup \{x = t\})$
  si $t \neq x$ et $x \notin FV(t)$
- **Check :** $(x \stackrel{?}{=} t \cup E, S) \Rightarrow fail$
  si $t \neq x$ et $x \in FV(t)$.