

Réécriture

Unification

dernière mise à jour: *22 février 2007 – 17 h 13*




Jacques Herbrand

Équation

Le but de l'unification est de résoudre des équations.

Un **problème équationnel** est un ensemble fini de paires ¹ de termes :

$$P = \{s_1 \stackrel{?}{=} t_1, \dots, s_n \stackrel{?}{=} t_n\}$$

¹Attention il s'agit bien de **paires** et pas de **couples** ! 


Équation

Le but de l'unification est de résoudre des équations.

Un **problème équationnel** est un ensemble fini de paires ¹ de termes :

$$P = \{s_1 \stackrel{?}{=} t_1, \dots, s_n \stackrel{?}{=} t_n\}$$

$s_i \stackrel{?}{=} t_i$ est une **équation**

¹Attention il s'agit bien de **paires** et pas de **couples** ! 

Convention

Si t_j dans $s_j \stackrel{?}{=} t_j$ est une variable x_j ,
on écrira plutôt $x_j \stackrel{?}{=} s_j$ que $s_j \stackrel{?}{=} x_j$!

La variable isolée, si elle existe est toujours à gauche.

Solution

Une **solution** de P ou un **unificateur** de P est une substitution σ telle que

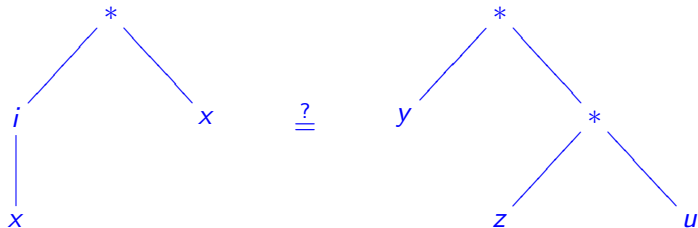
$$\begin{aligned}\sigma(s_1) &= \sigma(t_1) \\ &\vdots \\ \sigma(s_n) &= \sigma(t_n)\end{aligned}$$

L'**ensemble des unificateurs** de P est noté $\mathcal{U}(P)$.

Si $\mathcal{U}(P) \neq \emptyset$ on dit que P est **unifiable**.

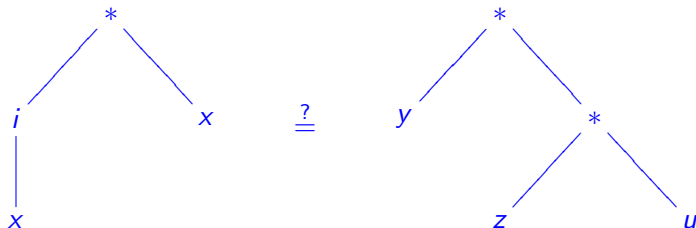
Exemple 1

Soit le problème :



Exemple 1

Soit le problème :

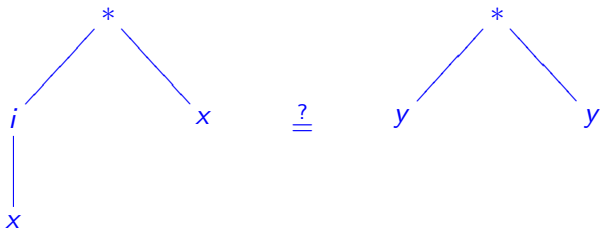


Les substitutions

$\{x \mapsto i(u) * u, \quad y \mapsto i(i(u) * u), \quad z \mapsto i(u)\}$
 $\{x \mapsto z * u, \quad y \mapsto i(z * u)\}$ sont chacune des solutions.

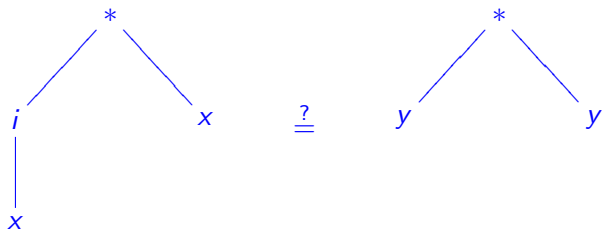
Exemple 2

Le problème :



Exemple 2

Le problème :



n'a pas de solution, car il faudrait que $\sigma(x) = i(\sigma(x))$.
Ce qui est impossible.

Ordre sur les substitutions

L'ensemble $Sub(T(\Sigma, V))$ est muni d'un ordre \lesssim défini par

$$\sigma \lesssim \tau \text{ si et seulement si } (\exists \rho \in Sub(T(\Sigma, V)) \quad \tau = \rho\sigma).$$

σ est dite **plus générale** que τ .

MGU

L'**unificateur le plus général** de P (ou **mgu**) est une substitution σ telle que

1. σ est un unificateur de P , c-à-d $\sigma \in \mathcal{U}(P)$.
2. Si $\theta \in \mathcal{U}(P)$ alors $\sigma \lesssim \theta$.

Forme résolue

Soit P un problème contenant $x \stackrel{?}{=} t$.

$x \stackrel{?}{=} t$ est en **forme résolue** dans P si x n'apparaît nulle part ailleurs dans P .

En particulier,

▶ $x \notin \text{Var}(t)$

▶ et il n'y a pas d'autre équation de la forme $x \stackrel{?}{=} s$.

x est alors dite **résolue**.

Un problème est résolu si toutes ses équations sont en forme résolue.

Si $P = \{x_1 \stackrel{?}{=} t_1, \dots, x_n \stackrel{?}{=} t_n\}$ est résolu,
alors $\sigma = \{x_1 \mapsto t_1, \dots, x_n \mapsto t_n\}$ est un mgu
de P .

On note σ_P

au lieu de $\{x_1 \mapsto t_1, \dots, x_n \mapsto t_n\}$

Si $P = \{x_1 \stackrel{?}{=} t_1, \dots, x_n \stackrel{?}{=} t_n\}$ est résolu,
alors $\sigma = \{x_1 \mapsto t_1, \dots, x_n \mapsto t_n\}$ est un mgu
de P .

Démonstration :

σ est un unificateur de P .

Pour $1 \leq i \leq n$, $\text{Var}(t_i) \cap \{x_1, \dots, x_n\} = \emptyset$, car P est résolu.

Il s'en suit que $\sigma(t_i) = t_i$, car en plus pour $y \notin \{x_1, \dots, x_n\}$

on a $\sigma(y) = y$.

Donc nous déduisons $\sigma(x_i) = t_i = \sigma(t_i)$, ce qui prouve que σ est un unificateur de P .

σ est minimal.

Soit τ un unificateur.

On a $\tau(x_i) = \tau(t_i)$ pour $1 \leq i \leq n$.

Donc aussi $\tau(x_i) = \tau(\sigma(x_i))$.

Si $y \notin \{x_1, \dots, x_n\}$, alors $\sigma(y) = y$ et $\tau(y) = \tau(\sigma(y))$.

Donc $\tau = \tau\sigma$, ce qui prouve que σ est un unificateur minimal (c'est-à-dire un mgu).

On notera qu'on a aussi montré que σ est le seul mgu, puisqu'en fait, tout unificateur est comparable à σ et plus «grand» que lui.

Transformations

Chaque transformation est écrite comme une règle d'inférence

$$R \frac{P_{\text{avant}}}{P_{\text{après}}}$$

Transformations

$$\text{Supprime} \quad \frac{\{u \stackrel{?}{=} u\} \cup P}{P}$$

$$\text{Decompose} \quad \frac{\{f(u_1, \dots, u_n) \stackrel{?}{=} f(v_1, \dots, v_n)\} \cup P}{\{u_1 \stackrel{?}{=} v_1, \dots, u_n \stackrel{?}{=} v_n\} \cup P} \quad f \in \Sigma_n$$

$$\text{Elimine} \quad \frac{\{x \stackrel{?}{=} v\} \cup P}{\{x \stackrel{?}{=} v\} \cup \sigma(P)}$$

où $x \stackrel{?}{=} v$ n'est pas en forme résolue dans $\{x \stackrel{?}{=} v\} \cup P$
et $x \notin \text{Var}(v)$ et $\sigma = \{x \mapsto v\}$.

Succès et échec

Si P est résolu, aucune règle ne s'applique (succès).

Ça n'est pas le seul cas d'arrêt. Les autres cas sont des échecs.

- ▶ S'il existe une équation de la forme

$f(u_1, \dots, u_m) \stackrel{?}{=} g(v_1, \dots, v_n)$ avec $f \neq g$, on dit qu'il y a clash.

- ▶ S'il existe une équation de la forme $x \stackrel{?}{=} v$ avec $x \in \text{Var}(v)$ on dit qu'il y a un cycle.

Résoudre par transformation

Si P' est obtenu à partir de P par application de l'une des règles, on écrit

$$P \Leftrightarrow P'$$

Résoudre par transformation

Si P' est obtenu à partir de P par application de l'une des règles, on écrit

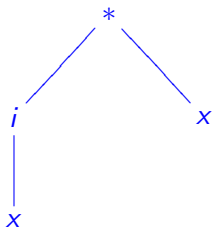
$$P \Leftrightarrow P'$$

Cette méthode à des liens avec l'élimination de Gauss.

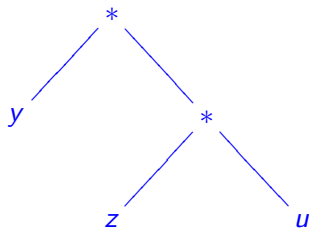
On simplifie par **Supprime** et **Decompose** et on élimine par **Elimine**.

Exercice

Appliquer les règles à :



?



Préservation des unificateurs : Si $P \Rightarrow P'$ alors $\mathcal{U}(P) = \mathcal{U}(P')$.

Préservation des unificateurs : Si $P \Rightarrow P'$ alors $\mathcal{U}(P) = \mathcal{U}(P')$.

La seule difficulté est **Elimine**.

Posons $P = \{s_1 \stackrel{?}{=} t_1, \dots, s_n \stackrel{?}{=} t_n\}$.

Supposons $x \stackrel{?}{=} v \cup P \Rightarrow x \stackrel{?}{=} v \cup \sigma(P)$

où $x \stackrel{?}{=} v$ n'est pas en forme résolue et $x \notin \text{Var}(v)$

et $\sigma = \{x \mapsto v\}$.

Soit τ telle que $\text{Dom}(\tau) \cap \text{Range}(\tau) = \emptyset$, alors $\tau\tau = \tau$,
on dit que τ est **idempotente**.

On voit que si les unificateurs idempotents sont préservés,
alors tous les unificateurs sont préservés ².

Soit τ idempotente telle que $\tau(x) = \tau(v)$,

- ▶ alors $\tau = \tau\sigma$,
- ▶ et dans le problème P , pour chaque i on a
 $\tau(s_i) = \tau\sigma(s_i) = \tau\sigma(t_i) = \tau(t_i)$.

Donc $\tau \in \mathcal{U}(x \stackrel{?}{=} v \cup P)$ si et seulement si $\tau \in \mathcal{U}(x \stackrel{?}{=} v \cup \sigma(P))$.

Par conséquent $\mathcal{U}(x \stackrel{?}{=} v \cup P) = \mathcal{U}(x \stackrel{?}{=} v \cup \sigma(P))$.

²En effet, un unificateur non idempotent est obtenu par renommage de variables à partir d'un unificateur idempotent.

Correction : Si $P \xrightarrow{*} P'$ et P' est résolu alors $mgu(P) = \sigma P'$.

Correction : Si $P \stackrel{*}{\Rightarrow} P'$ et P' est résolu alors $mgu(P) = \sigma_{P'}$.

On combine les deux lemmes précédents.

$\mathcal{U}(P) = \mathcal{U}(P')$ donc $mgu(P) = mgu(P')$.

Or $mgu(P') = \sigma_{P'}$.

c.q.f.d.

Complétude : Si P est unifiable alors il existe P'
tel que

▶ $P \stackrel{*}{\Rightarrow} P'$,

▶ P' est résolu.

Complétude : Si P est unifiable alors il existe P' tel que

- ▶ $P \stackrel{*}{\Rightarrow} P'$,
- ▶ P' est résolu.

Soit (P_i) une suite telle que $P_0 = P$ et $P_i \Rightarrow P_{i+1}$.

On veut montrer qu'il existe n tel que P_n est résolu.

Si P_i n'est pas résolu, alors

- ▶ si P_i ne contient pas de variable x non résolue, il contient une équation $s = t$ décomposable, alors **Decompose** s'applique.
- ▶ si P_i contient une équation triviale $x \stackrel{?}{=} x$ alors **Supprime** s'applique.
- ▶ si P_i contient une équation non résolue $x \stackrel{?}{=} t$ avec $x \notin \text{Var}(t)$, alors **Elimine** s'applique.

Complétude : Si P est unifiable alors il existe P' tel que

- ▶ $P \stackrel{*}{\Rightarrow} P'$,
- ▶ P' est résolu.

Soit (P_i) une suite telle que $P_0 = P$ et $P_i \Rightarrow P_{i+1}$.

On veut montrer qu'il existe n tel que P_n est résolu.

Si P_i n'est pas résolu, alors

- ▶ si P_i ne contient pas d'équation $x \stackrel{?}{=} v$ non résolue, il contient une équation $s = t$ décomposable, alors **Decompose** s'applique.
- ▶ si P_i contient une équation triviale $x \stackrel{?}{=} x$ alors **Supprime** s'applique.
- ▶ si P_i contient une équation non résolue $x \stackrel{?}{=} t$ avec $x \notin \text{Var}(t)$, alors **Elimine** s'applique.

Si P_i n'est pas résolu et est unifiable alors **le processus peut continuer**.

Complétude : Si P est unifiable alors il existe P' tel que

- ▶ $P \xrightarrow{*} P'$,
- ▶ P' est résolu.

Le processus s'arrête.

En effet, si l'on attache à chaque P_i le triplet (n_v, n_f, n_e) où

- ▶ n_v est le nombre de **variables non résolues**,
- ▶ n_f est le nombre de **symboles fonctionnels**,
- ▶ et n_e est le nombre d'**équations**.

Chaque règle **diminue** cette quantité suivant l'ordre lexicographique.

En effet,

La règle **Supprime**

- ▶ diminue le nombre de variables non résolues n_v (cas où la suppression de $x \stackrel{?}{=} x$ rend la variable x résolue),
- ▶ ou si elle ne diminue pas le nombre de variables non résolues, elle diminue le nombre n_f de symboles fonctionnels (cas où u contient au moins un symbole fonctionnel),
- ▶ sinon elle diminue le nombre d'équations n_e .

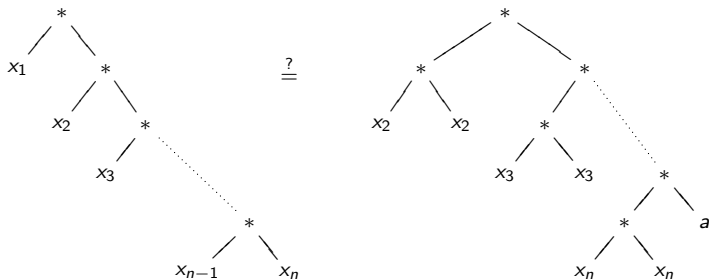
La règle **Decompose**

- ▶ diminue le nombre de variables non résolues n_v (cas où l'un des $u_i \stackrel{?}{=} v_j$ est de la forme $x \stackrel{?}{=} t$ et où x est résolue),
- ▶ ou si elle ne diminue pas le nombre de variables non résolues, elle diminue le nombre n_f de symboles fonctionnels.

La règle **Elimine**

- ▶ diminue le nombre n_v de variables non résolues.

L'unification peut être exponentielle en la taille du problème originel.



Par un succession de **Decompose** on obtient la (**forme triangulaire**)

$$\begin{aligned}x_1 &= x_2 * x_2 \\ &\vdots \\ x_{n-1} &= x_n * x_n \\ x_n &= a\end{aligned}$$

Puis par une succession de **Elimine**

$$x_1 = (\dots((a * a) * (a * a))\dots * ((a * a) * (a * a))\dots)$$

\vdots

$$x_{n-2} = (a * a) * (a * a)$$

$$x_{n-1} = (a * a)$$

$$x_n = a$$

Moralité

Pour une bonne efficacité, il faut utiliser la règle **Elimine** de façon paresseuse.

Il faut préférer

- ▶ une forme triangulaire

$$\begin{array}{l} x_1 = t(x_2, \dots, x_n) \\ \vdots \\ x_{n-1} = t(x_n) \\ x_n = t_n \end{array} \quad \text{Var}(t_n) = \emptyset$$

- ▶ ou une représentation par pointeurs
qui ne sont que des manières «indirectes»
de représenter les fonctions.