Unification

Unification

Two terms s and t are unifiable iff there exists a substitution (called unifier) s.t. $\theta(s) = \theta(t)$.

Example

- $f(x, g(x, a))$ and $f(h(a), y)$ are unifiable with $\theta = \{x/h(a), y/g(h(a), a)\}$

$$\begin{array}{ccc} f(x, g(x, a)) & \doteq & f(h(a), y) \\ f(h(a), g(h(a), a)) & = & f(h(a), g(h(a), a)) \end{array}$$

- $f(x, g(x, a))$ and $f(h(a), f(b, a))$ are not unifiable.

Principal substitutions

Let θ and τ be two substitutions and $\mathcal{S}$ be a set of substitutions.

- The composition of θ and τ is $(\theta \circ \tau)(x) = \widehat{\theta}(\tau(x))$ for every variable $x \in \mathcal{X}$.
- θ is an instance of τ (or τ is more general than θ) iff there exists a substitution ρ s.t. for every variable $x \in \mathcal{X}$, $(\rho \circ \tau)(x) = \theta(x)$.
- $\tau \in \mathcal{S}$ is principal (or most general) iff every substitution $\theta \in \mathcal{S}$ is an instance of τ.

Example

Let $\sigma_1 = \{y/b, z/h(c)\}$ and $\sigma_2 = \{x/f(y), y/z\}$.
$\sigma_1 \circ \sigma_2 = \{x/f(b), y/h(c), z/h(c)\}$.
σ_2 is more general than $\sigma_1 \circ \sigma_2$.

Principal unifier

Theorem

Let $\mathcal{S}$ be a non-empty set of unifiers of s and t. Then, there exists a principal unifier $\theta \in \mathcal{S}$ s.t. for every $\tau \in \mathcal{S}$, θ is more general than τ. Moreover, this principal unifier is unique modulo renaming.

Idempotent unifiers

A substitution θ is idempotent iff $\theta \circ \theta = \theta$.

Example

$\{y/b, z/h(c)\}$ is idempotent.
$\{x/f(y), y/z\}$ is not idempotent.

Theorem

If s and t are unifiable, then there exists a principal unifier of s and t which is idempotent.

How we can construct this unifier?

Equational systems

An (Σ)equational system is a set of (Σ)equations of the form $s \doteq t$.
An equational system E is unifiable iff there exists a unifier (called solution) for all the equations of E.

Finite equational systems are denoted $\{s_1 \doteq t_1, \ldots, s_n \doteq t_n\}$.

Solved forms

The equational system $E = \{s_1 \doteq t_1, \ldots, s_n \doteq t_n\}$ is in solved form iff

- All the s_i are distinct variables.
- No s_i appears in t_j.

Example

$E_0 = \{x \doteq y, z \doteq f(a)\}$ is in solved form but $E_1 = \{x \doteq y, x \doteq f(a)\}$,
$E_2 = \{x \doteq y, y \doteq f(a)\}$,
$E_3 = \{x \doteq z, y \doteq f(y)\}$ do not.

Notation : For the solved system $E = \{x_1 \doteq t_1, \ldots, x_n \doteq t_n\}$ we note $\vec{E}$ the substitution $\{x_1/t_1, \ldots, x_n/t_n\}$.

The transformation rules

$$\frac{E \cup \{s \doteq s\}}{E} \quad \text{(erase)} \qquad \frac{E \cup \{t \doteq x\} \qquad t \notin \mathcal{X}}{E \cup \{x \doteq t\}} \quad \text{(orient)}$$

$$\frac{E \cup \{f(s_1, \ldots, s_n) \doteq f(t_1, \ldots, t_n)\}}{E \cup \{s_1 \doteq t_1, \ldots, s_n \doteq t_n\}} \quad \text{(decompose)}$$

$$\frac{E \cup \{x \doteq s\} \qquad x \in Var(E) \qquad x \notin Var(s)}{E\{x \backslash s\} \cup \{x \doteq s\}} \quad \text{(substitute)}$$

The unification algorithm

1. Take an equational system E
2. Compute a new system P by applying the transformation rules as far as possible.
3. If the system P is in solved form
   - then send the answer $\vec{P}$
   - else fail

Example

Unification of the system $\{p(a, x, f(g(y))) \doteq p(z, f(z), f(u))\}$

$$\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{p(a,x,f(g(y))) \doteq p(z,f(z),f(u))}{a \doteq z, x \doteq f(z), f(g(y)) \doteq f(u)}\,d}{z \doteq a, x \doteq f(z), f(g(y)) \doteq f(u)}\,o}{z \doteq a, x \doteq f(a), f(g(y)) \doteq f(u)}\,s}{z \doteq a, x \doteq f(a), g(y) \doteq u}\,d}{z \doteq a, x \doteq f(a), u \doteq g(y)(solvedform)}\,o$$

yields the (idempotent) substitution $\{z/a, x/f(a), u/g(y)\}$.

Soundness and completeness of the algorithm

Theorem

The algorithm terminates.

Theorem

(Soundness) *If the algorithm finds a substitution $\vec{S}$ for the problem P, then P is unifiable and $\vec{S}$ is a m.g.u. of P.*
That is,
If P is not unifiable, then the algorithm fails.

Theorem

(Completeness) *If the system P is unifiable, then the algorithm computes the m.g.u. of P.*
That is,
If the algorithm fails, then the system P is not unifiable.