

Unification

basic definitions and results

Definition

A **unifier** of the terms s, t is a substitution σ such that $\sigma(s) = \sigma(t)$.

A substitution σ is **more general** than a substitution σ' if there is a substitution δ such that

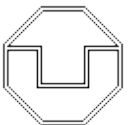
$$\sigma' = \delta\sigma.$$

In this case we write $\sigma \lesssim \sigma'$, and say that σ' is an **instance** of σ .

Definition

The unifier σ of the terms s, t is a **most general unifier (mgu)** iff every unifier of s, t is an instance of σ .

We will show: if s, t have a unifier, then they have an mgu, and this **mgu can effectively be computed**.



Unification

in more detail

Definition

A **unification problem** is a finite set of equations

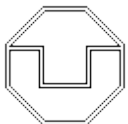
$$S = \{s_1 =? t_1, \dots, s_n =? t_n\}.$$

A **unifier** or **solution** of S is a substitution σ such that $\sigma(s_i) = \sigma(t_i)$ for $i = 1, \dots, n$.

$\mathcal{U}(S)$ denotes the **set of all unifiers** of S .

A substitution σ is a **most general unifier (mgu)** of S if

- $\sigma \in \mathcal{U}(S)$ and
- $\forall \sigma' \in \mathcal{U}(S). \sigma \lesssim \sigma'$.



Examples

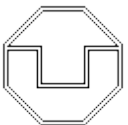
some typical situations

$\{f(x) =? f(a)\}$ has $\{x \mapsto a\}$ as a unifier

$\{x =? f(y)\}$ has $\{x \mapsto f(y)\}, \{x \mapsto f(a), y \mapsto a\}, \dots$ as unifiers

$\{x =? f(x)\}$ does **not** have a unifier

$\{f(x) =? g(y)\}$ does **not** have a unifier



Unification

by transformation

Idea:

Transform set of equations into solved form, from which the mgu can be obtained immediately.

Similar to Gaussian elimination in linear algebra:

$$\begin{array}{l} x + 3y = 0 \\ 2x + 8y = 2z \end{array}$$

\rightsquigarrow

$$\begin{array}{l} x + 3y = 0 \\ 2y = 2z \end{array}$$

\rightsquigarrow

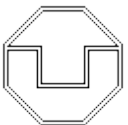
$$\begin{array}{l} x + 3y = 0 \\ y = z \end{array}$$

\rightsquigarrow

$$\begin{array}{l} x + 3z = 0 \\ y = z \end{array}$$

\rightsquigarrow

$$\begin{array}{l} x = -3z \\ y = z \end{array}$$



Definition

A unification problem

$$S = \{x_1 =? t_1, \dots, x_n =? t_n\}$$

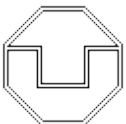
is in **solved form** if the x_i are pairwise distinct variables, none of which occurs in any of the t_i .

In this case we define

$$\vec{S} := \{x_1 \mapsto t_1, \dots, x_n \mapsto t_n\}.$$

Lemma

If S is in **solved form** then \vec{S} is an **mgu** of S .



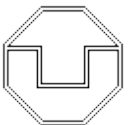
The transformation rules

Delete $\{t =? t\} \uplus S \implies S$

Decompose $\{f(t_1, \dots, t_n) =? f(u_1, \dots, u_n)\} \uplus S \implies \{t_1 =? u_1, \dots, t_n =? u_n\} \cup S$

Orient $\{t =? x\} \uplus S \implies \{x =? t\} \cup S$ if $t \notin V$

Eliminate $\{x =? t\} \uplus S \implies \{x =? t\} \cup \{x \mapsto t\}(S)$
if $x \in \mathcal{V}ar(S) - \mathcal{V}ar(t)$



Example

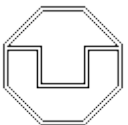
$$\{x =? f(a), g(x, x) =? g(x, y)\} \implies \text{Eliminate}$$

$$\{x =? f(a), g(f(a), f(a)) =? g(f(a), y)\} \implies \text{Decompose}$$

$$\{x =? f(a), f(a) =? f(a), f(a) =? y\} \implies \text{Delete}$$

$$\{x =? f(a), f(a) =? y\} \implies \text{Orient}$$

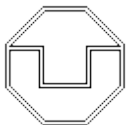
$$\{x =? f(a), y =? f(a)\}$$



$Unify(S)$ = `while` there is some T such that $S \implies T$ `do` $S := T$;
`if` S is in solved form `then` return \vec{S} `else` fail.

Lemma (termination, soundness, completeness)

1. *Unify* terminates for all inputs.
2. If $S \implies T$ then $\mathcal{U}(S) = \mathcal{U}(T)$.
3. If *Unify*(S) returns σ , then σ is an mgu of S .
4. If *Unify*(S) fails, then the final set of equations contains an equation of the form
 - (a) $x =? t$ with $x \in \mathcal{Var}(t)$, $x \neq t$,
 - (b) $f(\dots) =? g(\dots)$ with $f \neq g$.
5. If *Unify*(S) fails, then S has no solution.



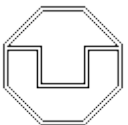
Theorem

The function *Unif* decides, for any input unification problem S , whether it has a solution or not.

If S has a solution, then *Unif* computes an mgu of S .

Complexity:

- The worst-case complexity of this unification algorithm is exponential (both time and space).
- There exists a linear time unification algorithm.



Matching

a special case of unification

Given terms l, s , find a substitution σ such that $\sigma(l) = s$.

The substitution σ is called a **matcher** of the **matching problem** $l \lesssim? s$.

Reduce matching to unification: regard all variables in s as constants, by introducing a new constant c_x for each variable x .

Example

The matching problem $f(x, y) \lesssim? f(g(z), x)$

becomes the unification problem $\{f(x, y) =? f(g(c_z), c_x)\}$.

The mgu $\{x \mapsto g(c_z), y \mapsto c_x\}$

becomes the matcher $\{x \mapsto g(z), y \mapsto x\}$.

