http://didawiki.di.unipi.it/doku.php/magistraleinformatica/psc/start

PSC 2023/24 (375AA, 9CFU)

Principles for Software Composition

Roberto Bruni
http://www.di.unipi.it/~bruni/

03 - Unification

Inference

SOS rule application?

1. a goal

$$(1 \oplus 2) \otimes (3 \oplus 4) \longrightarrow m$$

SOS rule application?

$$(prod)\,\frac{\mathsf{E}_0 \longrightarrow n_0 \qquad \mathsf{E}_1 \longrightarrow n_1}{\mathsf{E}_0 \otimes \mathsf{E}_1 \longrightarrow n}\; n = n_0 \cdot n_1$$

2. take a rule

$$(1 \oplus 2) \otimes (3 \oplus 4) \longrightarrow m$$

SOS rule application?

$$(prod)\,\frac{\mathsf{E}_0 \longrightarrow n_0 \qquad \mathsf{E}_1 \longrightarrow n_1}{\mathsf{E}_0 \otimes \mathsf{E}_1 \longrightarrow n}\quad n = n_0 \cdot n_1$$

$\mathsf{E}_0 = 1 \oplus 2$

$\mathsf{E}_1 = 3 \oplus 4$

$n = m$

3. unify
(if possible)

$$(1 \oplus 2) \otimes (3 \oplus 4) \longrightarrow m$$

SOS rule application?

$$(prod)\,\frac{1 \oplus 2 \longrightarrow n_0 \qquad 3 \oplus 4 \longrightarrow n_1}{(1 \oplus 2) \otimes (3 \oplus 4) \longrightarrow m}\; m = n_0 \cdot n_1$$

4. instantiate

$$\boxed{(1 \oplus 2) \otimes (3 \oplus 4) \longrightarrow m}$$

SOS rule application?

$$(prod)\frac{1 \oplus 2 \longrightarrow n_0 \quad 3 \oplus 4 \longrightarrow n_1}{(1 \oplus 2) \otimes (3 \oplus 4) \longrightarrow m}\ m = n_0 \cdot n_1$$

5. recursively solve subgoals

$$(1 \oplus 2) \otimes (3 \oplus 4) \longrightarrow m$$

SOS rule application?

$$(prod)\,\frac{1 \oplus 2 \longrightarrow \boxed{3} \quad 3 \oplus 4 \longrightarrow \boxed{7}}{(1 \oplus 2) \otimes (3 \oplus 4) \longrightarrow m}\; m = \boxed{3 \cdot 7}$$

6. combine results

$$(1 \oplus 2) \otimes (3 \oplus 4) \longrightarrow m$$

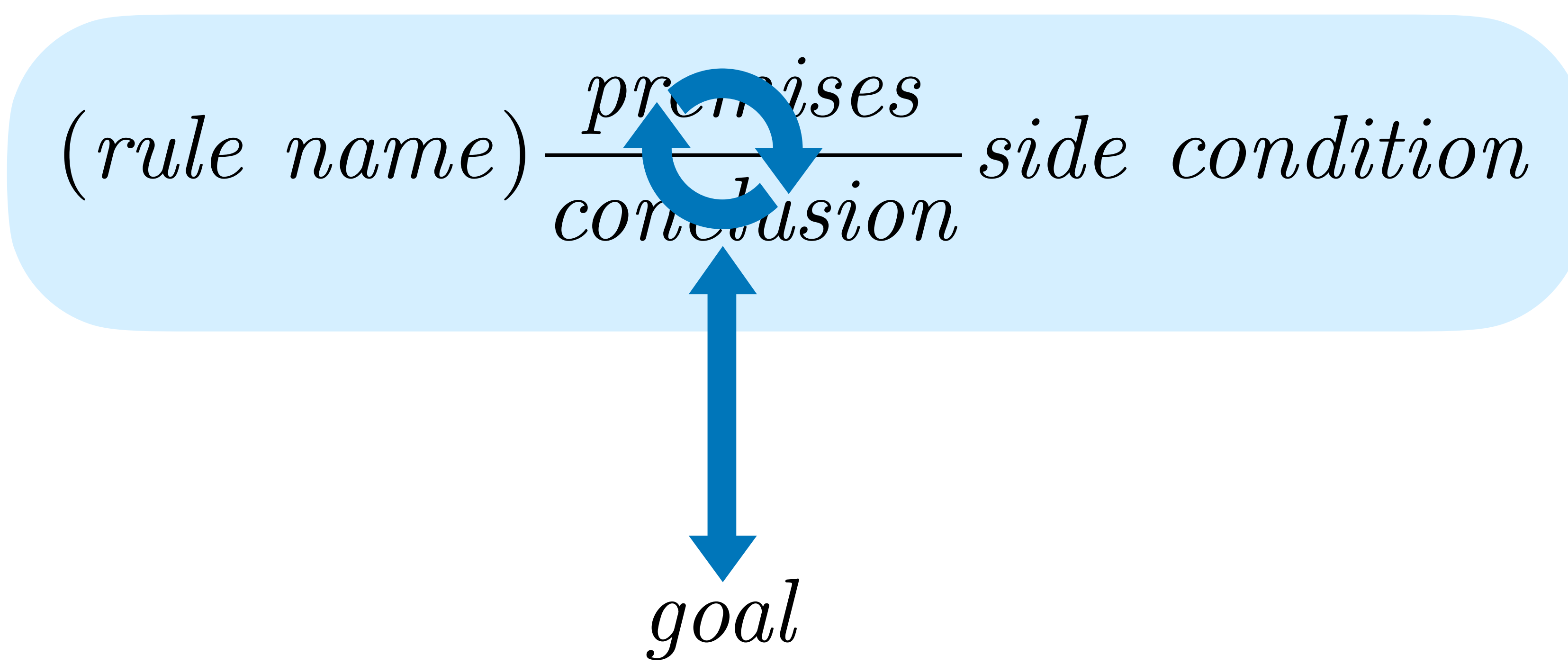
SOS rule application?

$$(prod)\,\frac{1 \oplus 2 \longrightarrow 3 \quad 3 \oplus 4 \longrightarrow 7}{(1 \oplus 2) \otimes (3 \oplus 4) \longrightarrow \boxed{21}}$$

7. return results

$$(1 \oplus 2) \otimes (3 \oplus 4) \longrightarrow \boxed{21}$$

Deduction process



Signatures

Unsorted signature

$$(\Sigma, ar)$$

a set of function symbols
(also called operators)

$$\Sigma = \{c, f, g, ...\}$$

arity function
(number of arguments)

$$ar : \Sigma \to \mathbb{N}$$

each function symbol has an arity

Running example

$\Sigma = \{0, \mathsf{succ}, \mathsf{plus}\}$

$ar(0) = 0$	constant
$ar(\mathsf{succ}) = 1$	unary
$ar(\mathsf{plus}) = 2$	binary

Equivalently

$$(\Sigma, ar)$$

let $\Sigma_n \triangleq ar^{-1}(n)$
$= \{f \in \Sigma \mid ar(f) = n\}$

a signature is an arity-indexed family of sets of operators

$$\Sigma = \{\Sigma_n\}_{n \in \mathbb{N}}$$

Running example

$$\Sigma_0 = \{0\}$$

$$\Sigma_1 = \{\mathsf{succ}\}$$

$$\Sigma_2 = \{\mathsf{plus}\}$$

$$\Sigma_n = \varnothing \qquad \text{if } n > 2$$

Terms over a signature

$\Sigma = \{\Sigma_n\}_{n \in \mathbb{N}}$ a signature

$X = \{x, y, z, ...\}$ an infinite set of variables

$T_{\Sigma,X}$ denotes the set of all terms over Σ, X

$T_{\Sigma,X}$ is the least set such that:

- if $x \in X$, then $x \in T_{\Sigma,X}$
- if $c \in \Sigma_0$, then $c \in T_{\Sigma,X}$
- if $f \in \Sigma_n$ and $t_1, ..., t_n \in T_{\Sigma,X}$, then $f(t_1, ..., t_n) \in T_{\Sigma,X}$

i.e. $$T_{\Sigma,X} \ni t ::= \underset{x \in X}{x} \mid \underset{c \in \Sigma_0}{c} \mid \underset{f \in \Sigma_n}{f(t_1, ..., t_n)}$$

Vars

$\Sigma = \{\Sigma_n\}_{n \in \mathbb{N}} \quad X = \{x, y, z, ...\} \qquad t \in T_{\Sigma,X}$

$vars(t)$ set of variables that appears in the term *t*

$$vars : T_{\Sigma,X} \to \wp(X)$$

$$vars(x) \triangleq \{x\}$$

$$vars(c) \triangleq \varnothing$$

$$vars(f(t_1, ..., t_n)) \triangleq \bigcup_{i=1}^{n} vars(t_i)$$

Closed terms

a term with no variables is called closed

$$T_\Sigma \triangleq vars^{-1}(\varnothing) = \{t \in T_{\Sigma,X} \mid vars(t) = \varnothing\}$$

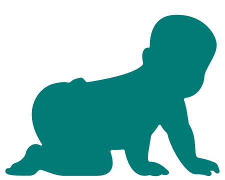
(obviously $T_\Sigma \subseteq T_{\Sigma,X}$)

Skill levels

advanced
multitasking

intermediate
does many things

novice
does one thing



beginner
does nothing

Running example

$$\Sigma_0 = \{\mathsf{0}\}$$

$$\Sigma_1 = \{\mathsf{succ}\}$$

$$\Sigma_2 = \{\mathsf{plus}\}$$

$$\Sigma_n = \varnothing \qquad \text{if } n > 2$$

Exercise

let's complete the schema
(obviously $T_\Sigma \subseteq T_{\Sigma,X}$)

t	$t \in ?$	$vars(t)$
0	(●) T_Σ (○) $T_{\Sigma,X}$	$\varnothing$
x	(○) T_Σ (●) $T_{\Sigma,X}$	$\{x\}$
succ(0)	(○) T_Σ (○) $T_{\Sigma,X}$	
succ(x)	(○) T_Σ (○) $T_{\Sigma,X}$	
succ(plus(0), x)	(○) T_Σ (○) $T_{\Sigma,X}$	
plus(succ(x), 0)	(○) T_Σ (○) $T_{\Sigma,X}$	
succ(succ(0), plus(x))	(○) T_Σ (○) $T_{\Sigma,X}$	
succ(plus(w, z))	(○) T_Σ (○) $T_{\Sigma,X}$	
plus(plus(x, succ(y)), plus(0, succ(x)))	(○) T_Σ (○) $T_{\Sigma,X}$	

Substitutions

$\rho : X \to T_{\Sigma,X}$ a substitution assigns terms to variables

we only consider substitutions that are identity everywhere, except for a finite number of cases, written

$$\rho = [x_1 = t_1\ , \ \dots\ , \ x_n = t_n]$$

all different

$$\rho(x) = \begin{cases} t_i & \text{if } x = x_i \\ x & \text{otherwise} \end{cases}$$

Notation

$\rho : X \to T_{\Sigma,X}$ a substitution assigns terms to variables

overloaded notation for the lifted function $\rho : T_{\Sigma,X} \to T_{\Sigma,X}$

$\rho(t)$ denotes the term obtained by simultaneous application of the substitution to all variable occurrences in *t*

$t\rho$ alternative notation

Example

$$\rho \triangleq [\, x = \mathsf{succ}(y) \;,\; y = \mathsf{0}]$$

$$t \triangleq \mathsf{plus}(\, \mathsf{plus}(x, y) \;,\; \mathsf{succ}(x) \,)$$

$$t\rho = \mathsf{plus}(\, \mathsf{plus}(\mathsf{succ}(y), \mathsf{0}) \;,\; \mathsf{succ}(\mathsf{succ}(y)) \,)$$

mgt relation

t is more general than t' if $\exists \rho.\ t' = t\rho$

in which case, we also say t' is an instance of t

$\mathsf{plus}(x, \mathsf{succ}(y))$	mgt	$\mathsf{plus}(0, \mathsf{succ}(\mathsf{succ}(z)))$
$\mathsf{plus}(0, x)$	~~mgt~~	$\mathsf{plus}(y, 0)$
$\mathsf{plus}(y, 0)$	~~mgt~~	$\mathsf{plus}(0, x)$
$\mathsf{plus}(0, x)$ $\mathsf{plus}(y, 0)$	mgt	$\mathsf{plus}(0, 0)$

mgt relation

mgt is transitive and reflexive

t mgt t

if (t_1 mgt t_2) and (t_2 mgt t_3), then (t_1 mgt t_3)

there are terms $t \neq t'$ such that (t mgt t') and (t' mgt t)

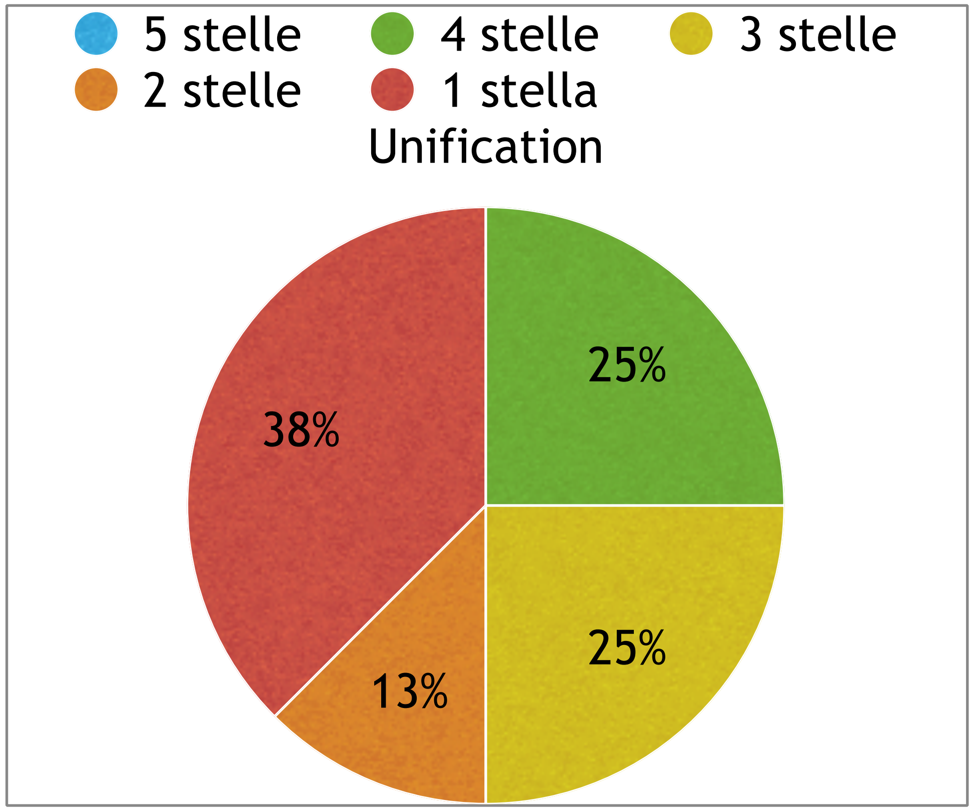
succ(x) succ(y)

mgt can be extended to substitutions pointwise

ρ mgt ρ' if $\exists \rho''.\ \forall x.\ \rho'(x) = \rho''(\rho(x))$

Unification
(in its simplest form: syntactic, first-order)

From your forms



(over 8 answers)

Unification problem

Given a set of potential equalities

$$\mathcal{G} = \{\ell_1 \stackrel{?}{=} r_1, ..., \ell_n \stackrel{?}{=} r_n\}$$

where $\ell_1, ..., \ell_n, r_1, ..., r_n \in T_{\Sigma,X}$

can we find a substitution ρ such that

$$\forall i \in [1,n].\ \rho(\ell_i) = \rho(r_i) \quad ?$$

we call such a ρ a solution of $\mathcal{G}$

$$sols(\mathcal{G}) \triangleq \{\rho \mid \forall i \in [1,n].\ \rho(\ell_i) = \rho(r_i)\}$$

Intuitively

Intuitively

Examples

$$\mathcal{G} = \{\mathsf{plus}(x, 0) \stackrel{?}{=} \mathsf{plus}(0, y)\}$$

$$sols(\mathcal{G}) \triangleq \{[x = 0, y = 0]\}$$

$$\mathcal{G} = \{\mathsf{succ}(x) \stackrel{?}{=} \mathsf{succ}(\mathsf{succ}(y))\}$$

$$sols(\mathcal{G}) \triangleq \{[x = \mathsf{succ}(y)], [x = \mathsf{succ}(0), y = 0], ...\}$$

Unification problem

More interestingly, can we solve the following problem?

Given a set of potential equalities

$$\mathcal{G} = \{\ell_1 \stackrel{?}{=} r_1, ..., \ell_n \stackrel{?}{=} r_n\}$$

where $\ell_1, ..., \ell_n, r_1, ..., r_n \in T_{\Sigma,X}$

can we find a most general solution ρ ?

$\rho \in sols(\mathcal{G})$ and

$\forall \rho' \in sols(\mathcal{G}).\ \rho$ mgt ρ'

Unification algorithm

Idea: we iteratively reduce the set $\mathcal{G}$
by solution-preserving tranformations until
either a solution is found
or we can prove there is no solution

$$\mathcal{G} \cdots \mathcal{G}_1 \cdots \mathcal{G}_n \cdots \{x_1 \stackrel{?}{=} t_1, ..., x_k \stackrel{?}{=} t_k\}$$

Solutions may not exist
and even if they exist may not be unique

Termination

$$\mathcal{G} = \{\ell_1 \stackrel{?}{=} r_1, \ldots, \ell_n \stackrel{?}{=} r_n\}$$

$\mathcal{G}$ and $\mathcal{G}'$ are equivalent if $sols(\mathcal{G}) = sols(\mathcal{G}')$

the algorithm terminates successfully when we reach

$\mathcal{G}' = \{x_1 \stackrel{?}{=} t_1, \ldots, x_k \stackrel{?}{=} t_k\}$ equivalent to $\mathcal{G}$

all different

$$\{x_1, \ldots, x_k\} \cap \bigcup_{i=1}^{k} vars(t_i) = \varnothing$$

any such $\mathcal{G}'$ defines a straightforward solution $[x_1 = t_1, \ldots, x_k = t_k]$

Notation

$$\mathcal{G} = \{\ell_1 \stackrel{?}{=} r_1, ..., \ell_n \stackrel{?}{=} r_n\}$$

$$vars(\mathcal{G}) \triangleq \bigcup_{i=1}^{n} (vars(\ell_i) \cup vars(r_i))$$

$$\mathcal{G}\rho \triangleq \{\ell_1\rho \stackrel{?}{=} r_1\rho, ..., \ell_n\rho \stackrel{?}{=} r_n\rho\}$$

Unification algorithm

delete

$\mathcal{G} \cup \{t \stackrel{?}{=} t\}$

becomes

$\mathcal{G}$

eliminate

$\mathcal{G} \cup \{x \stackrel{?}{=} t\}$

becomes if $x \in vars(\mathcal{G}) \setminus vars(t)$

$\mathcal{G}[x = t] \cup \{x \stackrel{?}{=} t\}$

swap

$\mathcal{G} \cup \{f(t_1, ..., t_m) \stackrel{?}{=} x\}$

becomes

$\mathcal{G} \cup \{x \stackrel{?}{=} f(t_1, ..., t_m)\}$

decompose

$\mathcal{G} \cup \{f(t_1, ..., t_m) \stackrel{?}{=} f(u_1, ..., u_m)\}$

becomes

$\mathcal{G} \cup \{t_1 \stackrel{?}{=} u_1, ..., t_m \stackrel{?}{=} u_m\}$

occur-check

$\mathcal{G} \cup \{x \stackrel{?}{=} f(t_1, ..., t_m)\}$

fails if $x \in vars(f(t_1, ..., t_m))$

conflict

$\mathcal{G} \cup \{f(t_1, ..., t_m) \stackrel{?}{=} g(u_1, ..., u_h)\}$

fails if $f \neq g$ or $m \neq h$

Example

$$\{\mathsf{plus}(\mathsf{succ}(x), x) \stackrel{?}{=} \mathsf{plus}(y, 0)\}$$

decompose

$$\mathcal{G} \cup \{f(t_1, ..., t_m) \stackrel{?}{=} f(u_1, ..., u_m)\}$$

becomes

$$\mathcal{G} \cup \{t_1 \stackrel{?}{=} u_1, ..., t_m \stackrel{?}{=} u_m\}$$

$$\{\mathsf{succ}(x) \stackrel{?}{=} y\ ,\ x \stackrel{?}{=} 0\}$$

eliminate

$$\mathcal{G} \cup \{x \stackrel{?}{=} t\}$$

becomes if $x \in vars(\mathcal{G}) \setminus vars(t)$

$$\mathcal{G}[x = t] \cup \{x \stackrel{?}{=} t\}$$

$$\{\mathsf{succ}(0) \stackrel{?}{=} y\ ,\ x \stackrel{?}{=} 0\}$$

swap

$$\mathcal{G} \cup \{f(t_1, ..., t_m) \stackrel{?}{=} x\}$$

becomes

$$\mathcal{G} \cup \{x \stackrel{?}{=} f(t_1, ..., t_m)\}$$

$$\{y \stackrel{?}{=} \mathsf{succ}(0)\ ,\ x \stackrel{?}{=} 0\}$$

success! $\rho = [y = \mathsf{succ}(0)\ ,\ x = 0]$

Example

$$\{\mathsf{plus}(0, x) \stackrel{?}{=} \mathsf{succ}(y)\}$$ plus $\neq$ succ

conflict

$$\mathcal{G} \cup \{f(t_1, ..., t_m) \stackrel{?}{=} g(u_1, ..., u_h)\}$$

fails if $f \neq g$ or $m \neq h$

failure!

Example

$$\{\mathsf{succ}(x) \stackrel{?}{=} y\,,\ \mathsf{succ}(y) \stackrel{?}{=} x\}$$

swap

$\mathcal{G} \cup \{f(t_1, ..., t_m) \stackrel{?}{=} x\}$

becomes

$\mathcal{G} \cup \{x \stackrel{?}{=} f(t_1, ..., t_m)\}$

$$\{\mathsf{succ}(x) \stackrel{?}{=} y\,,\ x \stackrel{?}{=} \mathsf{succ}(y)\}$$

eliminate

$\mathcal{G} \cup \{x \stackrel{?}{=} t\}$

becomes if $x \in vars(\mathcal{G}) \setminus vars(t)$

$\mathcal{G}[x = t] \cup \{x \stackrel{?}{=} t\}$

$$\{\mathsf{succ}(\mathsf{succ}(y)) \stackrel{?}{=} y\,,\ x \stackrel{?}{=} \mathsf{succ}(y)\}$$

swap

$\mathcal{G} \cup \{f(t_1, ..., t_m) \stackrel{?}{=} x\}$

becomes

$\mathcal{G} \cup \{x \stackrel{?}{=} f(t_1, ..., t_m)\}$

$$\{y \stackrel{?}{=} \mathsf{succ}(\mathsf{succ}(y))\,,\ x \stackrel{?}{=} \mathsf{succ}(y)\}$$

$y \in vars(\mathsf{succ}(\mathsf{succ}(y)))$

occur-check

$\mathcal{G} \cup \{x \stackrel{?}{=} f(t_1, ..., t_m)\}$

fails if $x \in vars(f(t_1, ..., t_m))$

failure!

Exercise

$$\{\mathsf{plus}(x, \mathsf{succ}(x)) \stackrel{?}{=} \mathsf{plus}(0, y)\ ,\ \mathsf{plus}(y, z) \stackrel{?}{=} \mathsf{plus}(z, w)\}$$

delete

$\mathcal{G} \cup \{t \stackrel{?}{=} t\}$
becomes
$\mathcal{G}$

eliminate

$\mathcal{G} \cup \{x \stackrel{?}{=} t\}$
becomes if $x \in vars(\mathcal{G}) \setminus vars(t)$
$\mathcal{G}[x = t] \cup \{x \stackrel{?}{=} t\}$

swap

$\mathcal{G} \cup \{f(t_1, \ldots, t_m) \stackrel{?}{=} x\}$
becomes
$\mathcal{G} \cup \{x \stackrel{?}{=} f(t_1, \ldots, t_m)\}$

decompose

$\mathcal{G} \cup \{f(t_1, \ldots, t_m) \stackrel{?}{=} f(u_1, \ldots, u_m)\}$
becomes
$\mathcal{G} \cup \{t_1 \stackrel{?}{=} u_1, \ldots, t_m \stackrel{?}{=} u_m\}$

occur-check

$\mathcal{G} \cup \{x \stackrel{?}{=} f(t_1, \ldots, t_m)\}$
fails if $x \in vars(f(t_1, \ldots, t_m))$

conflict

$\mathcal{G} \cup \{f(t_1, \ldots, t_m) \stackrel{?}{=} g(u_1, \ldots, u_h)\}$
fails if $f \neq g$ or $m \neq h$