

COSE212: Programming Languages

Lecture 17 — Automatic Type Inference (3)

Hakjoo Oh
2026 Fall

Finding a Solution of Type Equations

Find the values of type variables that make all the equations true.

$$\underbrace{\underbrace{\underbrace{\text{proc } (f)}_{t_f} \text{ proc } (x)}_{t_x} \underbrace{((f \ 3) - (f \ x))}_{t_2}}_{t_1} \underbrace{\hspace{10em}}_{t_0}$$

Diagram illustrating the structure of the type equations. The expression is a lambda abstraction $\text{proc } (f)$ applied to $\text{proc } (x)$ and the expression $((f \ 3) - (f \ x))$. The sub-expressions are grouped by braces and labeled with type variables: t_f for f , t_x for x , t_3 for $(f \ 3)$, t_4 for $(f \ x)$, t_2 for the subtraction $((f \ 3) - (f \ x))$, t_1 for the application $\text{proc } (f) \text{ proc } (x)$, and t_0 for the entire expression.

Equations	Solution
$t_0 = t_f \rightarrow t_1$	$t_0 = (\text{int} \rightarrow \text{int}) \rightarrow (\text{int} \rightarrow \text{int})$
$t_1 = t_x \rightarrow t_2$	$t_1 = \text{int} \rightarrow \text{int}$
$t_3 = \text{int}$	$t_2 = \text{int}$
$t_4 = \text{int}$	$t_3 = \text{int}$
$t_2 = \text{int}$	$t_4 = \text{int}$
$t_f = \text{int} \rightarrow t_3$	$t_f = \text{int} \rightarrow \text{int}$
$t_f = t_x \rightarrow t_4$	$t_x = \text{int}$

Static type systems find such a solution using *unification algorithm*.

Example 1

The calculation is split into equations to be solved and substitution found so far. Initially, the substitution is empty:

Equations		Substitution
t_0	$= t_f \rightarrow t_1$	
t_1	$= t_x \rightarrow t_2$	
t_3	$= \text{int}$	
t_4	$= \text{int}$	
t_2	$= \text{int}$	
t_f	$= \text{int} \rightarrow t_3$	
t_f	$= t_x \rightarrow t_4$	

Example 1

Consider each equation in turn. If the equation's left-hand side is a variable, move it to the substitution:

Equations	Substitution
$t_1 = t_x \rightarrow t_2$	$t_0 = t_f \rightarrow t_1$
$t_3 = \text{int}$	
$t_4 = \text{int}$	
$t_2 = \text{int}$	
$t_f = \text{int} \rightarrow t_3$	
$t_f = t_x \rightarrow t_4$	

Example 1

Move the next equation to the substitution and propagate the information to the existing substitution (i.e., substitute the right-hand side for each occurrence of t_1):

Equations	Substitution
$t_3 = \text{int}$	$t_0 = t_f \rightarrow (t_x \rightarrow t_2)$
$t_4 = \text{int}$	$t_1 = t_x \rightarrow t_2$
$t_2 = \text{int}$	
$t_f = \text{int} \rightarrow t_3$	
$t_f = t_x \rightarrow t_4$	

Example 1

Same for the next three equations:

Equations	Substitution
$t_4 = \text{int}$	$t_0 = t_f \rightarrow (t_x \rightarrow t_2)$
$t_2 = \text{int}$	$t_1 = t_x \rightarrow t_2$
$t_f = \text{int} \rightarrow t_3$	$t_3 = \text{int}$
$t_f = t_x \rightarrow t_4$	

Equations	Substitution
$t_2 = \text{int}$	$t_0 = t_f \rightarrow (t_x \rightarrow t_2)$
$t_f = \text{int} \rightarrow t_3$	$t_1 = t_x \rightarrow t_2$
$t_f = t_x \rightarrow t_4$	$t_3 = \text{int}$
	$t_4 = \text{int}$

Equations	Substitution
$t_f = \text{int} \rightarrow t_3$	$t_0 = t_f \rightarrow (t_x \rightarrow \text{int})$
$t_f = t_x \rightarrow t_4$	$t_1 = t_x \rightarrow \text{int}$
	$t_3 = \text{int}$
	$t_4 = \text{int}$
	$t_2 = \text{int}$

Example 1

Consider the next equation $t_f = \text{int} \rightarrow t_3$. The equation contains t_3 , which is already bound to int in the substitution. Substitute int for t_3 in the equation. This is called *applying* the substitution to the equation.

Equations	Substitution
$t_f = \text{int} \rightarrow \text{int}$	$t_0 = t_f \rightarrow (t_x \rightarrow \text{int})$
$t_f = t_x \rightarrow t_4$	$t_1 = t_x \rightarrow \text{int}$
	$t_3 = \text{int}$
	$t_4 = \text{int}$
	$t_2 = \text{int}$

Move the resulting equation to the substitution and update it.

Equations	Substitution
$t_f = t_x \rightarrow t_4$	$t_0 = (\text{int} \rightarrow \text{int}) \rightarrow (t_x \rightarrow \text{int})$
	$t_1 = t_x \rightarrow \text{int}$
	$t_3 = \text{int}$
	$t_4 = \text{int}$
	$t_2 = \text{int}$
	$t_f = \text{int} \rightarrow \text{int}$

Example 1

Apply the substitution to the equation:

Equations	Substitution
$\text{int} \rightarrow \text{int} = t_x \rightarrow \text{int}$	$t_0 = (\text{int} \rightarrow \text{int}) \rightarrow (t_x \rightarrow \text{int})$
	$t_1 = t_x \rightarrow \text{int}$
	$t_3 = \text{int}$
	$t_4 = \text{int}$
	$t_2 = \text{int}$
	$t_f = \text{int} \rightarrow \text{int}$

If neither side of the equation is a variable, simplify the equation by yielding two new equations:

Equations	Substitution
$\text{int} = t_x$	$t_0 = (\text{int} \rightarrow \text{int}) \rightarrow (t_x \rightarrow \text{int})$
$\text{int} = \text{int}$	$t_1 = t_x \rightarrow \text{int}$
	$t_3 = \text{int}$
	$t_4 = \text{int}$
	$t_2 = \text{int}$
	$t_f = \text{int} \rightarrow \text{int}$

Example 1

Switch the sides of the first equation and move it to the substitution:

Equations	Substitution
$\text{int} = \text{int}$	$t_0 = (\text{int} \rightarrow \text{int}) \rightarrow (\text{int} \rightarrow \text{int})$
	$t_1 = \text{int} \rightarrow \text{int}$
	$t_3 = \text{int}$
	$t_4 = \text{int}$
	$t_2 = \text{int}$
	$t_f = \text{int} \rightarrow \text{int}$
	$t_x = \text{int}$

The final substitution is the solution of the original equations.

Example 2

$$\underbrace{\text{proc } \underbrace{(f)}_{t_f} \underbrace{(f \ 11)}_{t_1}}_{t_0}$$

$$t_0 = t_f \rightarrow t_1$$

$$t_f = \text{int} \rightarrow t_1$$

Example 2

1

Equations	Substitution
$t_0 = t_f \rightarrow t_1$ $t_f = \text{int} \rightarrow t_1$	

2

Equations	Substitution
$t_f = \text{int} \rightarrow t_1$	$t_0 = t_f \rightarrow t_1$

3

Equations	Substitution
	$t_0 = (\text{int} \rightarrow t_1) \rightarrow t_1$ $t_f = \text{int} \rightarrow t_1$

The type is *polymorphic* in t_1 .

Example 3

if $\underbrace{x}_{t_x}$ then $\underbrace{(x - 1)}_{t_1}$ else 0

$\underbrace{\hspace{10em}}_{t_0}$

$t_x = \text{bool}$

$t_1 = t_0$

int = t_0

$t_x = \text{int}$

$t_1 = \text{int}$

Example 3

The equations have no solutions because, during the unification algorithm, we encounter the following contradictory state:

Equations	Substitution
$\text{bool} = \text{int}$	$t_x = \text{bool}$
$t_1 = \text{int}$	$t_1 = \text{int}$
	$t_0 = \text{int}$

Because `bool` and `int` cannot be equal, there is no solution to the equations.

Example 4

$$\text{proc } \underbrace{(f)}_{t_f} \underbrace{(\text{iszero } \underbrace{(f f)}_{t_2})}_{t_1}$$
$$\underbrace{\hspace{10em}}_{t_0}$$

$$t_0 = t_f \rightarrow t_1$$

$$t_1 = \text{bool}$$

$$t_2 = \text{int}$$

$$t_f = t_f \rightarrow t_2$$

Example 4

Solving as usual, we encounter a problem:

Equations	Substitution
$t_f = t_f \rightarrow \text{int}$	$t_0 = t_f \rightarrow \text{bool}$
	$t_1 = \text{bool}$
	$t_2 = \text{int}$

- There is no type t_f that satisfies the equation, because the right-hand side of the equation is always larger than the left.
- If we ever deduce an equation of the form $t = \dots t \dots$ where the type variable t occurs in the right-hand side, we must conclude that there is no solution. This is called *occurrence check*.

Unification Algorithm

For each equation in turn,

- Apply the current substitution to the equation.
- If the equation is always true (e.g. $\text{int} = \text{int}$), discard it.
- If the left- and right-hand sides are contradictory (e.g. $\text{bool} = \text{int}$), the algorithm fails.
- If neither side is a variable (e.g. $\text{int} \rightarrow t_1 = t_2 \rightarrow \text{bool}$), simplify the equation, which eventually generates an equation whose left- or right-hand side is a variable.
- If the left-hand side is not a variable, switch the sides.
- If the left-hand side variable occurs in the right-hand side, the algorithm fails.
- Otherwise, move it to the substitution and substitute the right-hand side for each occurrence of the variable in the substitution.

Exercise 1

`let  $x = 4$  in ( $x$  3)`

Exercise 2

let $f = \text{proc } (z) \ z \text{ in proc } (x) \ ((f \ x) - 1)$

Exercise 3

`let  $p$  = iszero 1 in if  $p$  then 88 else 99`

Exercise 4

```
let  $f$  = proc ( $x$ )  $x$  in if ( $f$  (iszero0)) then ( $f$  11) else ( $f$  22)
```

Substitution

Solutions of type equations are represented by substitution:

$$S \in \mathit{Subst} = \mathit{TyVar} \rightarrow T$$

Applying a substitution to a type:

$$\begin{aligned} S(\mathit{int}) &= \mathit{int} \\ S(\mathit{bool}) &= \mathit{bool} \\ S(\alpha) &= \begin{cases} t & \text{if } \alpha \mapsto t \in S \\ \alpha & \text{otherwise} \end{cases} \\ S(T_1 \rightarrow T_2) &= S(T_1) \rightarrow S(T_2) \end{aligned}$$

Example

Applying the substitution

$$S = \{t_1 \mapsto \text{int}, t_2 \mapsto \text{int} \rightarrow \text{int}\}$$

to to the type $(t_1 \rightarrow t_2) \rightarrow (t_3 \rightarrow \text{int})$:

$$\begin{aligned} & S((t_1 \rightarrow t_2) \rightarrow (t_3 \rightarrow \text{int})) \\ &= S(t_1 \rightarrow t_2) \rightarrow S(t_3 \rightarrow \text{int}) \\ &= (S(t_1) \rightarrow S(t_2)) \rightarrow (S(t_3) \rightarrow S(\text{int})) \\ &= (\text{int} \rightarrow (\text{int} \rightarrow \text{int})) \rightarrow (t_3 \rightarrow \text{int}) \end{aligned}$$

Unification

Update the current substitution with equality $t_1 \doteq t_2$.

unify : $T \times T \times Subst \rightarrow Subst$

unify(int, int, S) = S

unify(bool, bool, S) = S

unify(α , α , S) = S

unify(α , t , S) = $\begin{cases} \text{fail} & \alpha \text{ occurs in } t \\ \text{extend } S \text{ with } \alpha \doteq t & \text{otherwise} \end{cases}$

unify(t , α , S) = **unify**(α , t , S)

unify($t_1 \rightarrow t_2$, $t'_1 \rightarrow t'_2$, S) = let $S' = \text{unify}(t_1, t'_1, S)$ in
let $S'' = \text{unify}(S'(t_2), S'(t'_2), S')$ in
 S''

unify(-, -, -) = fail

Exercises

- $\text{unify}(\alpha, \text{int} \rightarrow \text{int}, \emptyset) =$
- $\text{unify}(\alpha, \text{int} \rightarrow \alpha, \emptyset) =$
- $\text{unify}(\alpha \rightarrow \beta, \text{int} \rightarrow \text{int}, \emptyset) =$
- $\text{unify}(\alpha \rightarrow \beta, \text{int} \rightarrow \alpha, \emptyset) =$
- $\text{unify}(\alpha \rightarrow \beta, \beta \rightarrow \text{int}, \emptyset) =$

Solving Equations

$$\mathbf{unifyall} : TyEqn \rightarrow Subst \rightarrow Subst$$

$$\mathbf{unifyall}(\emptyset, S) = S$$

$$\mathbf{unifyall}((t_1 \doteq t_2) \wedge u, S) = \text{let } S' = \mathbf{unify}(S(t_1), S(t_2), S) \\ \text{in } \mathbf{unifyall}(u, S')$$

Let $\mathcal{U}$ be the final unification algorithm:

$$\mathcal{U}(u) = \mathbf{unifyall}(u, \emptyset)$$

typeof : $E \rightarrow T$

The final type inference algorithm that composes equation derivation ($\mathcal{V}$) and equation solving ($\mathcal{U}$):

$$\begin{aligned} \mathbf{typeof}(E) = & \\ & \mathbf{let } S = \mathcal{U}(\mathcal{V}(\emptyset, E, \alpha)) \quad (\text{new } \alpha) \\ & \mathbf{in } S(\alpha) \end{aligned}$$

Examples

- **typeof**((proc (x) x) 1)
- **typeof**(let $x = 1$ in proc(y) ($x + y$))

Summary

Automatic type inference:

- derive type equations from the program text, and
- solve the equations by unification algorithm.