

Programming Languages and Compilers (CS 421)

Elsa L Gunter
2112 SC, UIUC

<http://courses.engr.illinois.edu/cs421>

Based in part on slides by Mattox Beckman, as updated by Vikram Adve and Gul Agha

10/19/22

1

Two Problems

- Type checking
 - Question: Does exp. e have type τ in env Γ ?
 - Answer: Yes / No
 - Method: Type **derivation**
- Typability
 - Question Does exp. e have **some type** in env. Γ ? If so, what is it?
 - Answer: Type τ / error
 - Method: Type **inference**

10/19/22

2

Type Inference - Outline

- Begin by assigning a type variable as the type of the whole expression
- Decompose the expression into component expressions
- Use typing rules to generate constraints on components and whole
- Recursively find substitution that solves typing judgment of first subcomponent
- Apply substitution to next subcomponent and find substitution solving it; compose with first, etc.
- Apply comp of all substitution to orig. type var. to get answer

10/19/22

3

Type Inference - Example

- What type can we give to
(fun x -> fun f -> f (f x))
- Start with a type variable and then look at the way the term is constructed

10/19/22

4

Type Inference - Example

- First approximate:
 $\{ \} \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha$
- Second approximate: use fun rule
$$\frac{\{x : \beta\} \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}{\{ \} \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}$$
- Remember constraint $\alpha \equiv (\beta \rightarrow \gamma)$

10/19/22

5

Type Inference - Example

- Third approximate: use fun rule
$$\frac{\frac{\{f : \delta ; x : \beta\} \vdash f (f x) : \varepsilon}{\{x : \beta\} \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}}{\{ \} \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}$$
- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

10/19/22

6

Type Inference - Example

- Fourth approximate: use app rule

$$\frac{\frac{\frac{\{f:\delta; x:\beta\}|- f : \varphi \rightarrow \varepsilon \quad \{f:\delta; x:\beta\}|- f x : \varphi}{\{f : \delta ; x : \beta\}|- (f (f x)) : \varepsilon}}{\{x : \beta\}|- (\text{fun } f \rightarrow f (f x)) : \gamma}}{\{ \}|- (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

10/19/22

7

Type Inference - Example

- Fifth approximate: use var rule, get constraint $\delta \equiv \varphi \rightarrow \varepsilon$, Solve with same
- Apply to next sub-proof

$$\frac{\frac{\frac{\{f:\delta; x:\beta\}|- f : \varphi \rightarrow \varepsilon \quad \{f:\delta; x:\beta\}|- f x : \varphi}{\{f : \delta ; x : \beta\}|- (f (f x)) : \varepsilon}}{\{x : \beta\}|- (\text{fun } f \rightarrow f (f x)) : \gamma}}{\{ \}|- (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

10/19/22

8

Type Inference - Example

- Current subst: $\{\delta \equiv \varphi \rightarrow \varepsilon\}$

$$\frac{\frac{\frac{\dots \quad \{f:\varphi \rightarrow \varepsilon; x:\beta\}|- f x : \varphi}{\{f : \delta ; x : \beta\}|- (f (f x)) : \varepsilon}}{\{x : \beta\}|- (\text{fun } f \rightarrow f (f x)) : \gamma}}{\{ \}|- (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

10/19/22

9

Type Inference - Example

- Current subst: $\{\delta \equiv \varphi \rightarrow \varepsilon\}$ Use App Rule

$$\frac{\frac{\frac{\{f:\varphi \rightarrow \varepsilon; x:\beta\}|- f:\zeta \rightarrow \varphi \quad \{f:\varphi \rightarrow \varepsilon; x:\beta\}|- x:\zeta}{\dots \quad \{f:\varphi \rightarrow \varepsilon; x:\beta\}|- f x : \varphi}}{\{f : \delta ; x : \beta\}|- (f (f x)) : \varepsilon}}{\{x : \beta\}|- (\text{fun } f \rightarrow f (f x)) : \gamma}}{\{ \}|- (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

10/19/22

10

Type Inference - Example

- Current subst: $\{\delta \equiv \varphi \rightarrow \varepsilon\}$
- Var rule: Solve $\zeta \rightarrow \varphi \equiv \varphi \rightarrow \varepsilon$ **Unification**

$$\frac{\frac{\frac{\{f:\varphi \rightarrow \varepsilon; x:\beta\}|- f:\zeta \rightarrow \varphi \quad \{f:\varphi \rightarrow \varepsilon; x:\beta\}|- x:\zeta}{\dots \quad \{f:\varphi \rightarrow \varepsilon; x:\beta\}|- f x : \varphi}}{\{f : \delta ; x : \beta\}|- (f (f x)) : \varepsilon}}{\{x : \beta\}|- (\text{fun } f \rightarrow f (f x)) : \gamma}}{\{ \}|- (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

10/19/22

11

Type Inference - Example

- Current subst: $\{\zeta \equiv \varepsilon, \varphi \equiv \varepsilon\} \circ \{\delta \equiv \varphi \rightarrow \varepsilon\}$
- Var rule: Solve $\zeta \rightarrow \varphi \equiv \varphi \rightarrow \varepsilon$ **Unification**

$$\frac{\frac{\frac{\{f:\varphi \rightarrow \varepsilon; x:\beta\}|- f:\zeta \rightarrow \varphi \quad \{f:\varphi \rightarrow \varepsilon; x:\beta\}|- x:\zeta}{\dots \quad \{f:\varphi \rightarrow \varepsilon; x:\beta\}|- f x : \varphi}}{\{f : \delta ; x : \beta\}|- (f (f x)) : \varepsilon}}{\{x : \beta\}|- (\text{fun } f \rightarrow f (f x)) : \gamma}}{\{ \}|- (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

10/19/22

12

Type Inference - Example

- Current subst: $\{\zeta \equiv \varepsilon, \varphi \equiv \varepsilon, \delta \equiv \varepsilon \rightarrow \varepsilon\}$
- Apply to next sub-proof

$$\frac{\dots \quad \frac{\{f : \varepsilon \rightarrow \varepsilon; x : \beta\} \vdash x : \varepsilon}{\dots \quad \{f : \varphi \rightarrow \varepsilon; x : \beta\} \vdash f x : \varphi}}{\{f : \delta; x : \beta\} \vdash (f (f x)) : \varepsilon}$$

$$\frac{\{x : \beta\} \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}{\{ \} \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

10/19/22

13

Type Inference - Example

- Current subst: $\{\zeta \equiv \varepsilon, \varphi \equiv \varepsilon, \delta \equiv \varepsilon \rightarrow \varepsilon\}$
- Var rule: $\varepsilon \equiv \beta$

$$\frac{\dots \quad \frac{\{f : \varepsilon \rightarrow \varepsilon; x : \beta\} \vdash x : \varepsilon}{\dots \quad \{f : \varphi \rightarrow \varepsilon; x : \beta\} \vdash f x : \varphi}}{\{f : \delta; x : \beta\} \vdash (f (f x)) : \varepsilon}$$

$$\frac{\{x : \beta\} \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}{\{ \} \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

10/19/22

14

Type Inference - Example

- Current subst: $\{\varepsilon \equiv \beta\} \circ \{\zeta \equiv \varepsilon, \varphi \equiv \varepsilon, \delta \equiv \varepsilon \rightarrow \varepsilon\}$
- Solves subproof; return one layer

$$\frac{\dots \quad \frac{\{f : \varepsilon \rightarrow \varepsilon; x : \beta\} \vdash x : \varepsilon}{\dots \quad \{f : \varphi \rightarrow \varepsilon; x : \beta\} \vdash f x : \varphi}}{\{f : \delta; x : \beta\} \vdash (f (f x)) : \varepsilon}$$

$$\frac{\{x : \beta\} \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}{\{ \} \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

10/19/22

15

Type Inference - Example

- Current subst: $\{\varepsilon \equiv \beta, \zeta \equiv \beta, \varphi \equiv \beta, \delta \equiv \beta \rightarrow \beta\}$
- Solves this subproof; return one layer

$$\frac{\dots \quad \frac{\{f : \varphi \rightarrow \varepsilon; x : \beta\} \vdash f x : \varphi}{\{f : \delta; x : \beta\} \vdash (f (f x)) : \varepsilon}}{\{x : \beta\} \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}$$

$$\frac{\{ \} \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}{\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)}$$

10/19/22

16

Type Inference - Example

- Current subst: $\{\varepsilon \equiv \beta, \zeta \equiv \beta, \varphi \equiv \beta, \delta \equiv \beta \rightarrow \beta\}$
- Need to satisfy constraint $\gamma \equiv (\delta \rightarrow \varepsilon)$, given subst, becomes: $\gamma \equiv ((\beta \rightarrow \beta) \rightarrow \beta)$

$$\frac{\dots \quad \frac{\{f : \delta; x : \beta\} \vdash (f (f x)) : \varepsilon}{\{x : \beta\} \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}}{\{ \} \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

10/19/22

17

Type Inference - Example

- Current subst: $\{\gamma \equiv ((\beta \rightarrow \beta) \rightarrow \beta), \varepsilon \equiv \beta, \zeta \equiv \beta, \varphi \equiv \beta, \delta \equiv \beta \rightarrow \beta\}$
- Solves subproof; return one layer

$$\frac{\dots \quad \frac{\{f : \delta; x : \beta\} \vdash (f (f x)) : \varepsilon}{\{x : \beta\} \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}}{\{ \} \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

10/19/22

18

Type Inference - Example

- Current subst:

$\{\gamma \equiv ((\beta \rightarrow \beta) \rightarrow \beta), \varepsilon \equiv \beta, \zeta \equiv \beta, \varphi \equiv \beta, \delta \equiv \beta \rightarrow \beta\}$

- Need to satisfy constraint $\alpha \equiv (\beta \rightarrow \gamma)$
given subst: $\alpha \equiv (\beta \rightarrow ((\beta \rightarrow \beta) \rightarrow \beta))$

$$\frac{\dots}{\{x : \beta\} \vdash (\text{fun } f \text{ -> } f (f x)) : \gamma}$$

$$\{ \} \vdash (\text{fun } x \text{ -> } \text{fun } f \text{ -> } f (f x)) : \alpha$$

- $\alpha \equiv (\beta \rightarrow \gamma)$;

10/19/22

19

Type Inference - Example

- Current subst:

$\{\alpha \equiv (\beta \rightarrow ((\beta \rightarrow \beta) \rightarrow \beta)),$

$\gamma \equiv ((\beta \rightarrow \beta) \rightarrow \beta), \varepsilon \equiv \beta, \zeta \equiv \beta, \varphi \equiv \beta, \delta \equiv \beta \rightarrow \beta\}$

- Solves subproof; return on layer

$$\frac{\dots}{\{x : \beta\} \vdash (\text{fun } f \text{ -> } f (f x)) : \gamma}$$

$$\{ \} \vdash (\text{fun } x \text{ -> } \text{fun } f \text{ -> } f (f x)) : \alpha$$

10/19/22

20

Type Inference - Example

- Current subst:

$\{\alpha \equiv (\beta \rightarrow ((\beta \rightarrow \beta) \rightarrow \beta)),$

$\gamma \equiv ((\beta \rightarrow \beta) \rightarrow \beta), \varepsilon \equiv \beta, \zeta \equiv \beta, \varphi \equiv \beta, \delta \equiv \beta \rightarrow \beta\}$

- Done: $\alpha \equiv (\beta \rightarrow ((\beta \rightarrow \beta) \rightarrow \beta))$

$$\{ \} \vdash (\text{fun } x \text{ -> } \text{fun } f \text{ -> } f (f x)) : \alpha$$

10/19/22

21

Type Inference Algorithm

Let $\text{infer}(\Gamma, e, \tau) = \sigma$

- Γ is a typing environment (giving polymorphic types to expression variables)
- e is an expression
- τ is a type (with type variables),
- σ is a substitution of types for type variables
- Idea: σ is substitution solving the constraints on type variables necessary for $\Gamma \vdash e : \tau$
- Should have $\sigma(\Gamma) \vdash e : \sigma(\tau)$ valid

10/19/22

22

Type Inference Algorithm

$\text{infer}(\Gamma, \text{exp}, \tau) =$

- Case *exp* of
 - Var $v \rightarrow$ return $\text{Unify}\{\tau \equiv \text{freshInstance}(\Gamma(v))\}$
 - Replace all quantified type vars by fresh ones
 - Const $c \rightarrow$ return $\text{Unify}\{\tau \equiv \text{freshInstance } \varphi\}$
where $\Gamma \vdash c : \varphi$ by the constant rules
 - fun $x \rightarrow e \rightarrow$
 - Let α, β be fresh variables
 - Let $\sigma = \text{infer}(\{x : \alpha\} + \Gamma, e, \beta)$
 - Return $\text{Unify}\{\{\sigma(\tau) \equiv \sigma(\alpha \rightarrow \beta)\}\} \circ \sigma$

10/19/22

23

Example of inference with Var Rule

Instance $\{a \rightarrow 'w\}$ ('w a fresh variable)

$\{x : \text{All } 'a. ('a * 'b) \text{ list}, y : \text{All. } 'b\} \vdash x : (\text{int} * \text{string}) \text{ list}$

$\text{freshInstance}(\text{All } 'a. ('a * 'b) \text{ list}) = ('w * 'b) \text{ list}$

$\text{Unify}\{((\text{int} * \text{string}) \text{ list} = ('w * 'b) \text{ list})\} = \{w \rightarrow \text{int}, 'b \rightarrow \text{string}\}$

After substitution:

Instance $\{a \rightarrow \text{int}\}$

$\{x : \text{All } 'a. ('a * \text{string}) \text{ list}, y : \text{All. string}\} \vdash x : (\text{int} * \text{string}) \text{ list}$

10/19/22

24

Type Inference Algorithm (cont)

- Case *exp* of
 - App ($e_1 e_2$) -->
 - Let α be a fresh variable
 - Let $\sigma_1 = \text{infer}(\Gamma, e_1, \alpha \rightarrow \tau)$
 - Let $\sigma_2 = \text{infer}(\sigma_1(\Gamma), e_2, \sigma_1(\alpha))$
 - Return $\sigma_2 \circ \sigma_1$

10/19/22

25

Type Inference Algorithm (cont)

- Case *exp* of
 - If e_1 then e_2 else e_3 -->
 - Let $\sigma_1 = \text{infer}(\Gamma, e_1, \text{bool})$
 - Let $\sigma_2 = \text{infer}(\sigma_1\Gamma, e_2, \sigma_1(\tau))$
 - Let $\sigma_3 = \text{infer}(\sigma_2 \circ \sigma_1(\Gamma), e_3, \sigma_2 \circ \sigma_1(\tau))$
 - Return $\sigma_3 \circ \sigma_2 \circ \sigma_1$

10/19/22

26

Type Inference Algorithm (cont)

- Case *exp* of
 - let $x = e_1$ in e_2 -->
 - Let α be a fresh variable
 - Let $\sigma_1 = \text{infer}(\Gamma, e_1, \alpha)$
 - Let $\sigma_2 = \text{infer}(\{x:\text{GEN}(\sigma_1(\Gamma), \sigma_1(\alpha))\} + \sigma_1(\Gamma), e_2, \sigma_1(\tau))$
 - Return $\sigma_2 \circ \sigma_1$

10/19/22

27

Type Inference Algorithm (cont)

- Case *exp* of
 - let rec $x = e_1$ in e_2 -->
 - Let α be a fresh variable
 - Let $\sigma_1 = \text{infer}(\{x:\alpha\} + \Gamma, e_1, \alpha)$
 - Let $\sigma_2 = \text{infer}(\{x:\text{GEN}(\sigma_1(\Gamma), \sigma_1(\alpha))\} + \sigma_1(\Gamma), e_2, \sigma_1(\tau))$
 - Return $\sigma_2 \circ \sigma_1$

10/19/22

28

Type Inference Algorithm (cont)

- To infer a type, introduce *type_of*
- Let α be a fresh variable
- *type_of* (Γ, e) =
 - Let $\sigma = \text{infer}(\Gamma, e, \alpha)$
 - Return $\sigma(\alpha)$
- Need an algorithm for *Unif*

10/19/22

29

Background for Unification

- **Terms** made from **constructors** and **variables** (for the simple first order case)
- Constructors may be **applied** to arguments (other terms) to make new terms
- Variables and constructors with no arguments are base cases
- Constructors applied to different number of arguments (**arity**) considered different
- **Substitution** of terms for variables

10/19/22

30

Simple Implementation Background

```
type term = Variable of string
          | Const of (string * term list)
let x = Variable "a";; let tm = Const ("2",[]);;

let rec subst var_name residue term =
  match term with Variable name ->
    if var_name = name then residue else term
  | Const (c, tys) ->
    Const (c, List.map (subst var_name residue)
                 tys);;
```

10/19/22

31

Unification Problem

Given a set of pairs of terms (“equations”)

$$\{(s_1, t_1), (s_2, t_2), \dots, (s_n, t_n)\}$$

(the *unification problem*) does there exist a substitution σ (the *unification solution*) of terms for variables such that

$$\sigma(s_i) = \sigma(t_i),$$

for all $i = 1, \dots, n$?

10/19/22

32

Uses for Unification

- Type Inference and type checking
- Pattern matching as in OCaml
 - Can use a simplified version of algorithm
- Logic Programming - Prolog
- Simple parsing

10/19/22

33

Unification Algorithm

- Let $S = \{(s_1 = t_1), (s_2 = t_2), \dots, (s_n = t_n)\}$ be a unification problem.
- Case $S = \{ \}$: $\text{Unif}(S) = \text{Identity function}$ (i.e., no substitution)
- Case $S = \{(s, t)\} \cup S'$: Four main steps

10/19/22

34

Unification Algorithm

- **Delete:** if $s = t$ (they are the same term) then $\text{Unif}(S) = \text{Unif}(S')$
- **Decompose:** if $s = f(q_1, \dots, q_m)$ and $t = f(r_1, \dots, r_m)$ (same f , same $m!$), then $\text{Unif}(S) = \text{Unif}(\{(q_1, r_1), \dots, (q_m, r_m)\} \cup S')$
- **Orient:** if $t = x$ is a variable, and s is not a variable, $\text{Unif}(S) = \text{Unif}(\{(x = s)\} \cup S')$

10/19/22

35

Unification Algorithm

- **Eliminate:** if $s = x$ is a variable, and x does not occur in t (the occurs check), then
 - Let $\varphi = \{x \rightarrow t\}$
 - $\text{Unif}(S) = \text{Unif}(\varphi(S')) \circ \{x \rightarrow t\}$
 - Let $\psi = \text{Unif}(\varphi(S'))$
 - $\text{Unif}(S) = \{x \rightarrow \psi(t)\} \circ \psi$
 - Note: $\{x \rightarrow a\} \circ \{y \rightarrow b\} = \{y \rightarrow (\{x \rightarrow a\}(b))\} \circ \{x \rightarrow a\}$ if y not in a

10/19/22

36

Tricks for Efficient Unification

- Don't return substitution, rather do it incrementally
- Make substitution be constant time
 - Requires implementation of terms to use mutable structures (or possibly lazy structures)
 - We won't discuss these

10/19/22

37

Example

- x, y, z variables, f, g constructors

- Unify $\{(f(x) = f(g(f(z), y))), (g(y, y) = x)\} = ?$

10/19/22

38

Example

- x, y, z variables, f, g constructors
- $S = \{(f(x) = f(g(f(z), y))), (g(y, y) = x)\}$ is nonempty

- Unify $\{(f(x) = f(g(f(z), y))), (g(y, y) = x)\} = ?$

10/19/22

39

Example

- x, y, z variables, f, g constructors
- Pick a pair: $(g(y, y) = x)$

- Unify $\{(f(x) = f(g(f(z), y))), (g(y, y) = x)\} = ?$

10/19/22

40

Example

- x, y, z variables, f, g constructors
- Pick a pair: $(g(y, y) = x)$
- Orient: $(x = g(y, y))$

- Unify $\{(f(x) = f(g(f(z), y))), (g(y, y) = x)\} =$
Unify $\{(f(x) = f(g(f(z), y))), (x = g(y, y))\}$
by Orient

10/19/22

41

Example

- x, y, z variables, f, g constructors

- Unify $\{(f(x) = f(g(f(z), y))), (x = g(y, y))\} = ?$

10/19/22

42

Example

- x, y, z variables, f, g constructors
- $\{(f(x) = f(g(f(z), y))), (x = g(y, y))\}$ is non-empty
- Unify $\{(f(x) = f(g(f(z), y))), (x = g(y, y))\} = ?$

10/19/22

43

Example

- x, y, z variables, f, g constructors
- Pick a pair: $(x = g(y, y))$
- Unify $\{(f(x) = f(g(f(z), y))), (x = g(y, y))\} = ?$

10/19/22

44

Example

- x, y, z variables, f, g constructors
- Pick a pair: $(x = g(y, y))$
- Eliminate x with substitution $\{x \rightarrow g(y, y)\}$
 - Check: x not in $g(y, y)$
- Unify $\{(f(x) = f(g(f(z), y))), (x = g(y, y))\} = ?$

10/19/22

45

Example

- x, y, z variables, f, g constructors
- Pick a pair: $(x = g(y, y))$
- Eliminate x with substitution $\{x \rightarrow g(y, y)\}$
- Unify $\{(f(x) = f(g(f(z), y))), (x = g(y, y))\} =$
Unify $\{(f(g(y, y)) = f(g(f(z), y)))\}$
 - $\{x \rightarrow g(y, y)\}$

10/19/22

46

Example

- x, y, z variables, f, g constructors
- Unify $\{(f(g(y, y)) = f(g(f(z), y)))\}$
 - $\{x \rightarrow g(y, y)\} = ?$

10/19/22

47

Example

- x, y, z variables, f, g constructors
- $\{(f(g(y, y)) = f(g(f(z), y)))\}$ is non-empty
- Unify $\{(f(g(y, y)) = f(g(f(z), y)))\}$
 - $\{x \rightarrow g(y, y)\} = ?$

10/19/22

48

Example

- x, y, z variables, f, g constructors
- Pick a pair: $(f(g(y, y)) = f(g(f(z), y)))$

- Unify $\{(f(g(y, y)) = f(g(f(z), y)))\}$
 - $\{x \rightarrow g(y, y)\} = ?$

10/19/22

49

Example

- x, y, z variables, f, g constructors
- Pick a pair: $(f(g(y, y)) = f(g(f(z), y)))$
- Decompose: $(f(g(y, y)) = f(g(f(z), y)))$ becomes $\{(g(y, y) = g(f(z), y))\}$

- Unify $\{(f(g(y, y)) = f(g(f(z), y)))\}$
 - $\{x \rightarrow g(y, y)\} =$
- Unify $\{(g(y, y) = g(f(z), y))\} \circ \{x \rightarrow g(y, y)\}$

10/19/22

50

Example

- x, y, z variables, f, g constructors
- $\{(g(y, y) = g(f(z), y))\}$ is non-empty

- Unify $\{(g(y, y) = g(f(z), y))\}$
 - $\{x \rightarrow g(y, y)\} = ?$

10/19/22

51

Example

- x, y, z variables, f, g constructors
- Pick a pair: $(g(y, y) = g(f(z), y))$

- Unify $\{(g(y, y) = g(f(z), y))\}$
 - $\{x \rightarrow g(y, y)\} = ?$

10/19/22

52

Example

- x, y, z variables, f, g constructors
- Pick a pair: $(f(g(y, y)) = f(g(f(z), y)))$
- Decompose: $(g(y, y) = g(f(z), y))$ becomes $\{(y = f(z)); (y = y)\}$

- Unify $\{(g(y, y) = g(f(z), y))\} \circ \{x \rightarrow g(y, y)\} =$
Unify $\{(y = f(z)); (y = y)\} \circ \{x \rightarrow g(y, y)\}$

10/19/22

53

Example

- x, y, z variables, f, g constructors

- Unify $\{(y = f(z)); (y = y)\} \circ \{x \rightarrow g(y, y)\} = ?$

10/19/22

54

Example

- x, y, z variables, f, g constructors
- $\{(y = f(z)); (y = y)\} \circ \{x \rightarrow g(y, y)\}$ is non-empty
- Unify $\{(y = f(z)); (y = y)\} \circ \{x \rightarrow g(y, y)\} = ?$

10/19/22

55

Example

- x, y, z variables, f, g constructors
- Pick a pair: $(y = f(z))$
- Unify $\{(y = f(z)); (y = y)\} \circ \{x \rightarrow g(y, y)\} = ?$

10/19/22

56

Example

- x, y, z variables, f, g constructors
- Pick a pair: $(y = f(z))$
- Eliminate y with $\{y \rightarrow f(z)\}$
- Unify $\{(y = f(z)); (y = y)\} \circ \{x \rightarrow g(y, y)\} =$
Unify $\{(f(z) = f(z))\}$
o $\{y \rightarrow f(z)\} \circ \{x \rightarrow g(y, y)\} =$
Unify $\{(f(z) = f(z))\}$
o $\{y \rightarrow f(z); x \rightarrow g(f(z), f(z))\}$

10/19/22

57

Example

- x, y, z variables, f, g constructors
- Unify $\{(f(z) = f(z))\}$
o $\{y \rightarrow f(z); x \rightarrow g(f(z), f(z))\} = ?$

10/19/22

58

Example

- x, y, z variables, f, g constructors
- $\{(f(z) = f(z))\}$ is non-empty
- Unify $\{(f(z) = f(z))\}$
o $\{y \rightarrow f(z); x \rightarrow g(f(z), f(z))\} = ?$

10/19/22

59

Example

- x, y, z variables, f, g constructors
- Pick a pair: $(f(z) = f(z))$
- Unify $\{(f(z) = f(z))\}$
o $\{y \rightarrow f(z); x \rightarrow g(f(z), f(z))\} = ?$

10/19/22

60

Example

- x, y, z variables, f, g constructors
- Pick a pair: $(f(z) = f(z))$
- Delete
- Unify $\{(f(z) = f(z))\}$
 - $\{y \rightarrow f(z); x \rightarrow g(f(z), f(z))\} =$
- Unify $\{\} \circ \{y \rightarrow f(z); x \rightarrow g(f(z), f(z))\}$

10/19/22

61

Example

- x, y, z variables, f, g constructors
- Unify $\{\} \circ \{y \rightarrow f(z); x \rightarrow g(f(z), f(z))\} = ?$

10/19/22

62

Example

- x, y, z variables, f, g constructors
- $\{\}$ is empty
- Unify $\{\} =$ identity function
- Unify $\{\} \circ \{y \rightarrow f(z); x \rightarrow g(f(z), f(z))\} =$
 $\{y \rightarrow f(z); x \rightarrow g(f(z), f(z))\}$

10/19/22

63

Example

- Unify $\{(f(x) = f(g(f(z), y))), (g(y, y) = x)\} =$
 $\{y \rightarrow f(z); x \rightarrow g(f(z), f(z))\}$

$$f(\quad x \quad) = f(g(f(z), y)) \\ \rightarrow f(g(f(z), f(z))) = f(g(f(z), f(z)))$$

$$g(y, y) = x \\ \rightarrow g(f(z), f(z)) = g(f(z), f(z))$$

10/19/22

64

Example of Failure: Decompose

- Unify $\{(f(x, g(y)) = f(h(y), x))\}$
- Decompose: $(f(x, g(y)) = f(h(y), x))$
- = Unify $\{(x = h(y)), (g(y) = x)\}$
- Orient: $(g(y) = x)$
- = Unify $\{(x = h(y)), (x = g(y))\}$
- Eliminate: $(x = h(y))$
- Unify $\{(h(y) = g(y))\} \circ \{x \rightarrow h(y)\}$
- No rule to apply! Decompose fails!

10/19/22

65

Example of Failure: Occurs Check

- Unify $\{(f(x, g(x)) = f(h(x), x))\}$
- Decompose: $(f(x, g(x)) = f(h(x), x))$
- = Unify $\{(x = h(x)), (g(x) = x)\}$
- Orient: $(g(x) = x)$
- = Unify $\{(x = h(x)), (x = g(x))\}$
- No rules apply.

10/19/22

66