

Лекция 2.

Задачи типизации λ_{\rightarrow} .

Выразительная сила λ_{\rightarrow} .

Основные задачи типизации λ -исчисления

Рассмотрим $? \vdash ? : ?$.

1. *Проверка типа*: выполняется ли $\Gamma \vdash M : \sigma$ для контекста Γ , терма M и типа σ
Компиляция в языке программирования, типизированном по Чёрчу. Проверка доказательства.

2. *Реконструкция типа*: $? \vdash M : ?$.

Компиляция в языке программирования, типизированном по Карри. Это бывает чаще, чем кажется.

```
template <class A, class B>  
auto min(A a, B b) -> decltype(a < b ? a : b) {  
    return (a < b) ? a : b;  
}
```

3. *Обитаемость типа*: $\Gamma \vdash ? : \sigma$.

Поиск доказательства.

Все задачи разрешимы.

Задача реконструкции типа

Определение

Алгебраический терм

$$\theta ::= x \mid (f \theta \dots \theta)$$

Определение

Подстановка переменных — функция $S_0 : V \rightarrow T$, где $S_0(x) = x$ почти везде (за исключением конечного множества переменных).

Подстановка: $S : T \rightarrow T$, что $S(x) = S_0(x)$, но $S(f \theta_1 \dots \theta_k) = f S(\theta_1) \dots S(\theta_k)$

$$S(\Gamma) = \{x : S(\tau_x) \mid x : \tau_x \in \Gamma\}$$

Определение

Будем воспринимать запись типа как некоторое выражение в алгебраических термах, импликация — единственный функциональный символ. Наиболее общей парой для задачи реконструкции типа $? \vdash M : ?$ назовём такие $\langle \Gamma, \gamma \rangle$, что:

1. $\Gamma \vdash M : \gamma$
2. Если $\Delta \vdash M : \delta$, то найдётся такая подстановка S , что $\Delta = S(\Gamma)$ и $\delta = S(\gamma)$.

Общий план решения

1. Основа решения — алгоритм унификации для системы уравнений в алгебраических термах.
2. По терму M строим систему уравнений в алгебраических термах.
3. Наиболее общим унификатором системы будет является подстановка, из которой можно получить наиболее общую пару.

Система уравнений в алгебраических термах

Определение

Система уравнений в алгебраических термах

$$\begin{cases} \theta_1 = \sigma_1 \\ \vdots \\ \theta_n = \sigma_n \end{cases}$$

где θ_i и σ_i — термы

Задача унификации

Определение

Решением задачи унификации для системы уравнений $\sigma_k = \tau_k$ назовём такую подстановку S , что $S(\sigma_k) = S(\tau_k)$.

Определение

Наиболее общим решением задачи унификации назовём такую подстановку S , что для любого другого решения T найдётся подстановка R , что $T(\rho) = R(S(\rho))$.

Определение

Система в разрешённой форме — каждое уравнение имеет вид $x_i = \theta_i$, причём каждый из x_i входит в систему ровно один раз (является левой частью одного из уравнений)

Определение

Система несовместна — система не имеет решений.

Алгоритм унификации

Пусть дана система уравнений $\sigma_i = \tau_i$. Возьмём произвольное уравнение и попробуем проверить/применить одно из следующих условий/действий к нему:

- (a) $\sigma_i = x$ если σ_i не переменная перепишем как $x = \sigma_i$
- (b) $\sigma_i = \sigma_i$ удалим
- (c) $f \theta_1 \dots \theta_n = f \rho_1 \dots \rho_n$ заменим на n уравнений $\theta_k = \rho_k$
- (d) если уравнение имеет вид $x = \tau_i$ и x входит хотя бы в одно другое уравнение, то заменим все другие уравнения на $\sigma_k[x := \tau_i] = \tau_k[x := \tau_i]$
- (e) если уравнение имеет вид $x = f \dots x_i \dots$, система несовместна (occurs check)
- (f) если уравнение имеет вид $f \dots = g \dots$ при $f \neq g$, система несовместна.

Если нет ни одного подходящего правила ни для одного уравнения — закончим работу (система находится в разрешённой форме).

Алгоритм всегда завершает работу

- ▶ Рассмотрим $\langle x, y, z \rangle$, где:
 - ▶ x — количество переменных, входящих в систему, которые входят не в разрешённом виде. Переменная t входит в систему в разрешённом виде, если переменная входит в систему ровно один раз, причём входит в уравнение вида $t = \sigma$;
 - ▶ y — количество функциональных символов в системе;
 - ▶ z — количество уравнений типа $a = a$ и $\theta = b$, где θ не переменная.
- ▶ Упорядочим тройки лексикографически (согласно порядковому типу ω^3).
- ▶ Заметим, что операции (a) и (b) всегда уменьшают z и иногда уменьшают x . Операция (c) всегда уменьшает y , иногда x и, возможно, увеличивает z . Операция (d) всегда уменьшает x и иногда увеличивает y . То есть, операции $(a) - (d)$ всегда уменьшают соответствующий ординал.
- ▶ Согласно лемме из матлога любая строго убывающая последовательность ординалов имеет конечную длину.

Корректность алгоритма

Теорема

Для системы уравнений $\sigma_k = \tau_k$ алгоритм даёт наиболее общее решение, если оно существует.

Доказательство.

- ▶ Операции (a) — (d) не меняют множества решений системы. За конечное время либо выполнится условие (e) или (f), либо будут исчерпаны правила.
- ▶ Условия (e), (f) очевидно означают несовместность системы (в т.ч. исходной).
- ▶ При отсутствии возможности применения правил и условий все уравнения имеют вид $x = \theta_x$, где x входит в систему только один раз. Построим $S_0(x) = \theta_x$.
- ▶ Если есть подстановка $T : T(\sigma_k) = T(\tau_k)$, тогда положим $R = \mathcal{U}(\{S_0(x) = T_0(x) \mid x \neq T_0(x)\})$. Очевидно, $T(\zeta) = R(S(\zeta))$



Построение системы по терму M

Будем строить систему рекурсией по структуре терма M (предполагаем, что все имена для связанных переменных уникальны). Каждой переменной x сопоставим свежую типовую переменную α_x . Также каждой аппликации $P\ Q$ в терме сопоставим свежую типовую переменную β_{PQ} .

По терму M и по всем его подтермам рекурсивно построим пару $\langle \mathcal{E}_M, \sigma_M \rangle$ так:

$$\langle \mathcal{E}_M, \sigma_M \rangle := \begin{cases} \langle \emptyset, \alpha_x \rangle, & M = x \\ \langle \mathcal{E}_P, \alpha_x \rightarrow \sigma_P \rangle, & M = \lambda x. P \\ \langle \mathcal{E}_P \cup \mathcal{E}_Q \cup \{ \sigma_P = \sigma_Q \rightarrow \beta_{PQ} \}, \beta_{PQ} \rangle, & M = P\ Q \end{cases}$$

Теорема

Если $S = \mathcal{U}(\mathcal{E}_M)$, то наиболее общим решением задачи типизации будет $\langle \{x : S(\alpha_x) \mid x \in FV(M)\}, S(\sigma_M) \rangle$

Доказательство.

Индукция по структуре M .



Пример вывода типов

1. Выберем пример ($M = \lambda f. \lambda x. f (f x)$) и индуктивно составим систему:

- ▶ Для $f x$: $\langle \{ \alpha_f = \alpha_x \rightarrow \beta_{fx} \}, \beta_{fx} \rangle$
- ▶ Для $f (f x)$: $\langle \{ \alpha_f = \alpha_x \rightarrow \beta_{fx}, \alpha_f = \beta_{fx} \rightarrow \beta_{ffx} \}, \beta_{ffx} \rangle$
- ▶ Для $\lambda x. f (f x)$: $\langle \{ \alpha_f = \alpha_x \rightarrow \beta_{fx}, \alpha_f = \beta_{fx} \rightarrow \beta_{ffx} \}, \alpha_x \rightarrow \beta_{ffx} \rangle$
- ▶ Для $\lambda f. \lambda x. f (f x)$: $\langle \{ \alpha_f = \alpha_x \rightarrow \beta_{fx}, \alpha_f = \beta_{fx} \rightarrow \beta_{ffx} \}, \alpha_f \rightarrow \alpha_x \rightarrow \beta_{ffx} \rangle$

2. Приводим систему к разрешённой форме:

- ▶ $\alpha_f = \alpha_x \rightarrow \beta_{fx}, \alpha_f = \beta_{fx} \rightarrow \beta_{ffx}$
- ▶ $\alpha_f = \alpha_x \rightarrow \beta_{fx}, \alpha_x \rightarrow \beta_{fx} = \beta_{fx} \rightarrow \beta_{ffx}$, правило (d)
- ▶ $\alpha_f = \alpha_x \rightarrow \beta_{fx}, \alpha_x = \beta_{fx}, \beta_{fx} = \beta_{ffx}$, правило (c)
- ▶ $\alpha_f = \beta_{fx} \rightarrow \beta_{fx}, \alpha_x = \beta_{fx}, \beta_{fx} = \beta_{ffx}$, правило (d)
- ▶ $\alpha_f = \beta_{ffx} \rightarrow \beta_{ffx}, \alpha_x = \beta_{ffx}, \beta_{fx} = \beta_{ffx}$, правило (d)

3. Строим функцию подстановки: $S_0(\alpha_f) = \beta_{ffx} \rightarrow \beta_{ffx}$, $S_0(\alpha_x) = S_0(\beta_{fx}) = \beta_{ffx}$

4. Наиболее общая пара: $\langle \emptyset, S(\alpha_f \rightarrow \alpha_x \rightarrow \beta_{ffx}) \rangle$, то есть

$$\vdash \lambda f. \lambda x. f (f x) : (\beta_{ffx} \rightarrow \beta_{ffx}) \rightarrow \beta_{ffx} \rightarrow \beta_{ffx}$$

Проверка типа

1. Задача реконструкции типа находит наиболее общую типизацию.
2. Сведём задачу проверки $\Gamma \vdash M : \sigma$ к задаче реконструкции типа $? \vdash M : ?$ и найдём $\langle \Gamma', \sigma' \rangle$.
3. Проверим, является ли $\langle \Gamma, \sigma \rangle$ частным случаем $\langle \Gamma', \sigma' \rangle$.

Обитаемость типа

1. Задача поиска M , что $\Gamma \vdash M : \sigma$.
2. Эквивалентно поиску доказательства утверждения σ в ИИП (разрешимо).
3. По доказательству затем получим его краткую запись в виде терма.

Выразительная сила

Определение

Расширенный полином, где $P(x)$, $P(x, y)$ — полиномы (выражения, составленные из сложения, умножения, аргументов и натуральных констант), c — константа:

$$E(m, n) := \begin{cases} c, & m = 0, n = 0 \\ P_1(m), & n = 0 \\ P_2(n), & m = 0 \\ P_3(m, n), & m > 0, n > 0 \end{cases}$$

Теорема

Пусть $\eta = (\alpha \rightarrow \alpha) \rightarrow (\alpha \rightarrow \alpha)$. Если $F : \eta \rightarrow \eta \rightarrow \eta$, то найдётся такой расширенный полином $E(m, n)$, что при всех $m, n \in \mathbb{N}_0$ выполнено $F \overline{m} \overline{n} =_{\beta} \overline{E(m, n)}$, либо $F \overline{m} \overline{n} =_{\beta} \lambda f.f$ при $E(m, n) = 1$.

Расширение языка: полное ИИВ

- ▶ Попробуем увеличить выразительную силу, воспользовавшись изоморфизмом Карри-Ховарда. Рассмотрим полное ИИВ.
- ▶ Расширим язык:

$$\begin{array}{ll} \Lambda ::= & x \mid (\Lambda \ \Lambda) \mid (\lambda x. \Lambda) \\ & \mid \langle \Lambda, \Lambda \rangle \mid (\pi_L \ \Lambda) \mid (\pi_R \ \Lambda) & \text{термы для } \& \\ & \mid (In_L \ \Lambda) \mid (In_R \ \Lambda) \mid (Case \ \Lambda \ \Lambda \ \Lambda) & \text{термы для } \vee \\ & \mid (Absurd \ \Lambda) & \text{термы для } \perp \end{array}$$

Новые связки требуют отдельных правил

- Упорядоченная пара в бестиповом лямбда-исчислении.

$$\text{MkPair} ::= \lambda a. \lambda b. \underbrace{(\lambda p. p \ a \ b)}_{\text{MkPair } a \ b} \quad \text{Fst} ::= \lambda p. p \ T \quad \text{Snd} ::= \lambda p. p \ F$$

- Какой тип у $\text{MkPair } a \ b$?

$$\text{MkPair } a \ b = \lambda p^{\alpha \rightarrow \beta \rightarrow \gamma}. p \ a^{\alpha} \ b^{\beta} : (\alpha \rightarrow \beta \rightarrow \gamma) \rightarrow \gamma$$

- Тип зависит от типа результата γ : при левой проекции $\alpha = \gamma$

$$\text{Fst} = \lambda p. p^{(\alpha \rightarrow \beta \rightarrow \gamma) \rightarrow \gamma} T^{\alpha \rightarrow \beta \rightarrow \alpha} : \gamma$$

При правой проекции $\beta = \gamma$: $\text{Snd} = \lambda p. p^{(\alpha \rightarrow \beta \rightarrow \gamma) \rightarrow \gamma} F^{\alpha \rightarrow \beta \rightarrow \beta} : \gamma$

- Как известно, связки в ИИВ друг через друга не выражаются. Поэтому никакая формула не сможет типизировать упорядоченную пару. Однако, в данном варианте типизации может помочь квантор по γ или схема аксиом (правил вывода).

Дополнительные правила для расширенного языка

1. Типизация дизъюнкции (алгебраического типа)

$$\frac{\Gamma \vdash A : \varphi}{\Gamma \vdash \text{In}_L A : \varphi \vee \psi} \quad \frac{\Gamma \vdash B : \psi}{\Gamma \vdash \text{In}_R B : \varphi \vee \psi}$$
$$\frac{\Gamma \vdash L : \varphi \vee \psi, \quad \Gamma \vdash f : \varphi \rightarrow \tau, \quad \Gamma \vdash g : \psi \rightarrow \tau}{\Gamma \vdash \text{Case } L f g : \tau}$$

2. Типизация конъюнкции (упорядоченной пары)

$$\frac{\Gamma \vdash A : \varphi \quad \Gamma \vdash B : \psi}{\Gamma \vdash \langle A, B \rangle : \varphi \& \psi} \quad \frac{\Gamma \vdash P : \varphi \& \psi}{\Gamma \vdash \text{Fst } P : \varphi} \quad \frac{\Gamma \vdash P : \varphi \& \psi}{\Gamma \vdash \text{Snd } P : \psi}$$

3. Типизация лжи

$$\frac{\Gamma \vdash A : \perp}{\Gamma \vdash \text{Absurd } A : \varphi}$$