1 Лекция 5
Изоморфизм Карри-Ховарда (завершение), Унификация

1.1 Изоморфизм Карри-Ховарда

Определение 1.1. Изоморфизм Карри-Ховарда

1. $\Gamma \vdash M : \sigma$ влечет $|\Gamma| \vdash \sigma$
2. $\Gamma \vdash \sigma$, то существует M и существует Δ, такое что $|\Delta|$=Γ, что $\Delta \vdash$M:σ, где Δ=$\{x_\sigma$:$\sigma | \sigma \in \Gamma\}$

Пример.

{f:$\alpha \to \beta$, x:β}$\vdash$fx:β
Применив изоморфизм Карри-Ховарда получим: $\{\alpha \to \beta, \beta\} \vdash \beta$

Доказательство. □

П.1 доказывается индукцией по длине выражения т.е. есть 3 правила вывода. убирая P и Q.

П.2 доказывается аналогичным способом но действия обратные.
Т.е. отношения между типами в системе типов могут рассматриваться как образ отношений между высказываниями в логической системе, и наоборот.

Определение 1.2. Расширенный полином определяется формулой:

$$E(p,q) = \begin{cases} C, & \text{if } p = q = 0 \\ p_1(p), & \text{if } q = 0 \\ p_2(q), & \text{if } p = 0 \\ p_3(p,q), & \text{if } p, q \neq 0 \end{cases}$$

, где C – константа, p_1, p_2, p_3 – выражения, составленные из $*, +, p, q$ и констант

по сути расширенный полином это множество функций над натуральными числами(черчевскими нумералами).

Пусть $\upsilon = (\alpha \to \alpha) \to (\alpha \to \alpha)$, где α–произвольный тип и пусть $F \in \Lambda$, что $F : \upsilon \to \upsilon \to \upsilon$, то существует расширенный полином E, такой что $\forall a, b \in \mathbb{N}\ F(\overline{a}, \overline{b}) =_\beta \overline{E(a,b)}$, где $\overline{a}$–черчевский нумерал

Теорема 1.1. У каждого терма в просто типизиреумом λ исчислении существует расширенный полином.

Утверждение 1.1. Основные задачи типизации λ исчисления

1. *Проверка типа*–выполняется ли $\Gamma \vdash M : \sigma$ для контекста Γ терма M и типа σ (для проверки типа обычно откидывают σ и рассматривают п.2).
2. *Реконструкция типа*–можно ли подставить вместо ? и $?_1$ в $?_1 \vdash M$:? подставить конкретный тип σ в ? и контекст Γ в $?_1$.

3. *Обитаемость типа*—пытается подобрать, такой **замкнутый** терм M и контекст Γ, что бы было выполнено $\Gamma \vdash M : \sigma$.

Определение 1.3. Алгебраический терм

Выражение типа $\Theta = a|(f_k\Theta_1 \cdots \Theta_n)$,
где a—переменная, $(f_k\Theta_1 \cdots \Theta_n)$—применение функции

Пример.

1. $(fab(ga))$
2. Известно, что $\to$ —функция, тогда выражение $((a \to b) \to c) \Longleftrightarrow (\to (\to ab)c)$

1.2 Уравнение в алгебраических термах $\Theta_1 = \Theta_2$ Система уравнений в алгебраических термах

Определение 1.4. 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

Определение 1.5. $\{a_i\} = A$—множество перменных, $\{\Theta_i\} = T$—множество термов.

Определение 1.6. Подстановка—отображение вида: $S_0 : A \to T$, которое является решением в алгебраических термах.

Т.е. S_0—конечное множество переменных $a_1 \cdots a_n$ на которых $S_0(a_i) = \Theta_i$ либо $S_0(a_i) = a_i$.

Доопределим S на все T т.е. $S : T \to T$, где

1. $S(a) = S_0(a)$
2. $S(f(\Theta_1 \cdots \Theta_k)) = f(S(\Theta_1) \cdots S(\Theta_k))$

По сути S тоже самое что и много if'ов либо map строк

Определение 1.7. Решить уравнение в алгебраических термах—найти такое S, что $S(\Theta_1) = S(\Theta_2)$

Пример.

Заранее обозначим: a, b – переменные, f, g, h – функции

1. $f(a(gb)) = f(he)d$ имеет решение $S(a) = he$ и $S(d) = gb$
   (a) $S(fa(gb)) = f(he)(gb)$
   (b) $S(f(he)d) = f(he)(gb)$
   (c) $f(he)(gb) = f(he)(gb)$
2. $fa = gb$—решений не имеет

Таким образом, что бы существовало решение необходимо равенство строк полученной подстановки

1.3 Алгоритм Унификации

1. Система уравнений E_1 эквивалентна E_2, если они имеют одинаковые решения(унификаторы).
2. Любая система E эквивалентна нектрому уравнению $\Sigma_1 = \Sigma_2$.

   Доказательство. □

   Возьмем функциональный символ f, не использующийся в E,

   $$E = \begin{cases} \Theta_1 = \sigma_1 \\ \vdots \\ \Theta_n = \sigma_n \end{cases}$$

   это же уравнение можно записать как$- f\Theta_1 \dots \Theta_n = f\sigma_1 \dots \sigma_n$

   Если сущесвтует подстановка S такая, что

   $S(\Theta_i) = S(\sigma_i)\, \forall i$, то $S(f\ \Theta_1 \dots \Theta_n) = f\ S(\sigma_1) \dots S(\sigma_n)$

   Обратное аналогично.

3. Рассмотрим операции

   (a) *Редукция терма*
   Заменим уравнение вида$-f_1\ \Theta_1 \dots \Theta_n = f_1\ \sigma_1 \dots \sigma_n$ на систему уравнений
   $\Theta_1 = \sigma_1$
   $\vdots$
   $\Theta_n = \sigma_n$

   (b) *Устранение переменной*
   Пусть есть уравнеие $x = \Theta$, заменим во всех остальных уравнениях переменную x на терм Θ

   Утверждение 1.2. Эти операции не изменяют множества решений.

Определение 1.8. Система уравнений в разрешеной форме если

1. Все уравнения имеют вид $a_i = \Theta_i$
2. Каждый из a_i входит в систему уравнений только раз

Определение 1.9. Система несовместима если

1. существует уравнение вида $f\ \Theta_1 \dots \Theta_n = g\ \sigma_1 \dots \sigma_n$, где $f \neq g$
2. существует уравнение вида $a = f\ \Theta_1 \dots \Theta_n$, причем a выходит в какой-то из Θ_i

1.4 Алгоритм унификации

1. Пройдемся по системе, выберем такое уравнение, что оно удовлетворяет одному из условий:
   (a) Если $\Theta_i = a_i$, то перепишем, как $a_i = \Theta_i$, Θ_i−не переменная
   (b) $a_i = a_i$− удалим
   (c) $f\,\Theta_1 \dots \Theta_n = f\,\sigma_1 \dots \sigma_n$ применим редукцию термов
   (d) $a_i = \Theta_i$ Применим подстановку переменной т.е. подставим во все остальне уравнения Θ_i вместо a_i
2. Проверим разрешима ли система, совместима ли система (два пункта несовместимости)
3. повторим пункт 1

Утверждение 1.3. Алгоритм не изменяет множетва решений

Утверждение 1.4. Несовместимая решения не имеет решений

Утверждение 1.5. Система в разрешеной форме имеет решение:

$$\begin{cases} a_1 = \Theta_1 \\ \vdots \\ a_n = \Theta_n \end{cases} \quad \text{имеет решение}- \quad \begin{cases} S_0(a_1) = \Theta_1 \\ \vdots \\ S_0(a_n) = \Theta_n \end{cases}$$

Утверждение 1.6. Алгоритм всегда закначивается

Доказательство. □

По индукции, выберем три числа $\langle x\, y\, z \rangle$, где
x−количество переменных, которые встречаются строго больше одного раза в левой части некоторого уравнения (т.е. b не повлияет на x, а a повлияет в уравнении $f(a(ga)b) = \Theta$),
y− количество функциональных символов в системе,
z−количество уравнеий типа $a = a$ и $\Theta = b$
Заметим, что (a) и (b) всегда уменьшают z и иногда уменьшают x,
(c) всегда уменьшает y иногда x и, возможно, увеличивает z,
операция (d) всегда уменьшает x, и иногда увеличивает y.
Очевидно, что с каждой операцией $a-d$ данная тройка уменьшается и так как $x, y, z \geq 0$, то данный алгоритм завершится за конечное время.

Пример.

Исходная система
$$E =$$
$$g(x_2) = x_1$$
$$f(x_1, h(x_1), x_2) = f(g(x_3), x_4, x_3)$$

Применим пункт (c) ко второму уравнению верхней системы получим:

$$\begin{aligned} E = \\ g(x_2) = x_1 \\ x_1 = g(x_3) \\ h(x_1) = x_4 \\ x_2 = x_3 \end{aligned}$$

Применим пункт (d) ко второму уравнению верхней системы (оно изменит 1ое уравнение) получим:

$$\begin{aligned} E = \\ g(x_2) = g(x_3) \\ x_1 = g(x_3) \\ h(g(x_3)) = x_4 \\ x_2 = x_3 \end{aligned}$$

Применим пункт (c) ко первому ур-ию и пункт (a) к третьему уравнению верхней системы

$$\begin{aligned} E = \\ x_2 = x_3 \\ x_1 = g(x_3) \\ x_4 = h(g(x_3)) \\ x_2 = x_3 \end{aligned}$$

Применим пункт (b) к последнему уравнению и получим систему в разрешеной форме

$$\begin{aligned} E = \\ x_2 = x_3 \\ x_1 = g(x_3) \\ x_4 = h(g(x_3)) \end{aligned}$$

Решение системы:

$$\begin{aligned} S = \{ \\ (x_1 = g(x_3)), \\ (x_2 = x_3), \\ (x_4 = h(g(x_3)))) \\ \} \end{aligned}$$

Определение 1.10. $S \circ T-$композиция подстановок, если $S \circ T = S(T(a))$

Определение 1.11. $S-$наиболее общий унификатор ксли любое решение сиситмы R может быть получено уточнением: $\exists T : R = T \circ S$

Утверждение 1.7. Алгоритм дает наиболее общий унификатор системы, если у нее есть решения. Если решений нет алгоритм окончится неудачей.