Системы типизации лямбда-исчисления

Лекция 5. Свойства просто типизированной системы

Денис Москвин

13.03.2011

CS Club при ПОМИ РАН

Система $\lambda\!\to$ а ля Карри (по-пуритански)

Предтермы:	Редукция:
$\Lambda ::= \quad V = \{a, b, \ldots\}$ $\quad\mid\ \Lambda\,\Lambda$ $\quad\mid\ \lambda V.\,\Lambda$	$(\lambda V.\,\Lambda_1)\,\Lambda_2 \to_\beta \Lambda_1[V := \Lambda_2]$
Типы:	**Типизация:**
$\mathbb{T} ::= \quad \mathbb{V} = \{\alpha, \beta, \ldots\}$ $\quad\mid\ \mathbb{T}\to\mathbb{T}$ **Контексты:** $\Gamma ::= \quad \varnothing$ $\quad\mid\ \Gamma, V{:}\mathbb{T}$	$\dfrac{V{:}\mathbb{T} \in \Gamma}{\Gamma \vdash V{:}\mathbb{T}}$ $\dfrac{\Gamma \vdash \Lambda_1{:}\mathbb{T}_1\to\mathbb{T}_2 \quad \Gamma \vdash \Lambda_2{:}\mathbb{T}_1}{\Gamma \vdash (\Lambda_1\,\Lambda_2){:}\mathbb{T}_2}$ $\dfrac{\Gamma, V{:}\mathbb{T}_1 \vdash \Lambda{:}\mathbb{T}_2}{\Gamma \vdash (\lambda V.\,\Lambda){:}\mathbb{T}_1\to\mathbb{T}_2}$

Система $\lambda_\to$ а ля Карри (конвенционально)

Предтермы:	Редукция:
$\Lambda ::= V \mid M\,N \mid \lambda x.\,M$	$(\lambda x.\,M)\,N \to_\beta M[x := N]$
Типы: $\mathbb{T} ::= \mathbb{V} \mid \sigma \to \tau$ Контексты: $\Gamma ::= \varnothing \mid \Gamma, x : \sigma$	Типизация: $\dfrac{x : \sigma \in \Gamma}{\Gamma \vdash x : \sigma}$ $\dfrac{\Gamma \vdash M : \sigma \to \tau \quad \Gamma \vdash N : \sigma}{\Gamma \vdash (M\,N) : \tau}$ $\dfrac{\Gamma, x : \sigma \vdash M : \tau}{\Gamma \vdash (\lambda x.\,M) : \sigma \to \tau}$

Здесь $V = \{a, b, \ldots\}$, $\mathbb{V} = \{\alpha, \beta, \ldots\}$ и $x \in V$; $M, N \in \Lambda$; $\sigma, \tau \in \mathbb{T}$.

Система $\lambda{\to}$ а ля Чёрч (конвенционально)

Предтермы:	Редукция:
$\Lambda_{\mathbb{T}} ::= V \mid M\,N \mid \lambda x : \sigma .\, M$	$(\lambda x : \sigma .\, M)\, N \to_\beta M[x := N]$
Типы: $\mathbb{T} ::= \mathbb{V} \mid \sigma \to \tau$ Контексты: $\Gamma ::= \varnothing \mid \Gamma, x : \sigma$	Типизация: $\dfrac{x : \sigma \in \Gamma}{\Gamma \vdash x : \sigma}$ $\dfrac{\Gamma \vdash M : \sigma \to \tau \quad \Gamma \vdash N : \sigma}{\Gamma \vdash (M\,N) : \tau}$ $\dfrac{\Gamma, x : \sigma \vdash M : \tau}{\Gamma \vdash (\lambda x : \sigma .\, M) : \sigma \to \tau}$

Здесь $V = \{a, b, \ldots\}$, $\mathbb{V} = \{\alpha, \beta, \ldots\}$ и $x \in V$; $M, N \in \Lambda_{\mathbb{T}}$; $\sigma, \tau \in \mathbb{T}$.

Свойства $\lambda_{\rightarrow}$: Лемма об инверсии

Лемма об инверсии [TAPL] (генерации [LCWT])

- $\Gamma \vdash x : \sigma \;\Rightarrow\; (x : \sigma) \in \Gamma$.
- $\Gamma \vdash (M\,N) : \tau \;\Rightarrow\; \exists \sigma\; [\Gamma \vdash M : \sigma \rightarrow \tau \;\wedge\; \Gamma \vdash N : \sigma]$.
- $\Gamma \vdash (\lambda x.\, M) : \rho \;\Rightarrow\; \exists \sigma, \tau\; [\Gamma, x : \sigma \vdash M : \tau \;\wedge\; \rho \equiv \sigma \rightarrow \tau]$. ($\lambda_{\rightarrow}$ а ля Карри)
- $\Gamma \vdash (\lambda x : \sigma.\, M) : \rho \;\Rightarrow\; \exists \tau\; [\Gamma, x : \sigma \vdash M : \tau \;\wedge\; \rho \equiv \sigma \rightarrow \tau]$. ($\lambda_{\rightarrow}$ а ля Чёрч)

Доказательство. Непосредственно из правил типизации.

Типизируемость подтерма

Пусть M' — подтерм M. Тогда $\Gamma \vdash M : \sigma \;\Rightarrow\; \Gamma' \vdash M' : \sigma'$ для некоторых Γ' и σ'. То есть, если терм имеет тип, то каждый его подтерм тоже имеет тип.

Доказательство. Индукциия по генерации $M : \sigma$.

Свойства $\lambda{\rightarrow}$: контексты

Какой контекст требуется, чтобы произвести присваивание типов?

Пусть $\Gamma = \{x_1{:}\sigma_1, \ldots, x_n{:}\sigma_n\}$ есть контекст.

▶ Будем писать $\mathrm{dom}(\Gamma) = \{x_1, \ldots, x_n\}$, $\sigma_i = \Gamma(x_i)$.
(Γ рассматривается как частичная функция.)

▶ Пусть V_0 — множество переменных. Обозначим ***сужение*** контекста до этого множества

$$\Gamma \restriction V_0 = \{x{:}\sigma \mid x \in V_0 \ \wedge \ \sigma = \Gamma(x)\}$$

Леммы о контекстах

Пусть Γ и Δ — контексты, причём $\Delta \supseteq \Gamma$. Тогда:

- $\Gamma \vdash M : \sigma \;\Rightarrow\; \Delta \vdash M : \sigma$. Thinning, расширение контекста не влияет на выводимость утверждения типизации.
- $\Gamma \vdash M : \sigma \;\Rightarrow\; FV(M) \subseteq \mathrm{dom}(\Gamma)$. Свободные переменные типизированного терма должны присутствовать в контексте.
- $\Gamma \vdash M : \sigma \;\Rightarrow\; \Gamma \upharpoonright FV(M) \vdash M : \sigma$. Сужение контекста до множества свободных переменных терма не влияет на выводимость утверждения типизации.

Доказательство. Индукциия по выводу $M{:}\sigma$.

Свойства $\lambda\!\to$: нетипизируемые предтермы

Рассмотрим предтерм $x\,x$. Предположим, что это терм. Тогда имеются Γ и σ, такие что

$$\Gamma \vdash (x\,x):\sigma$$

По лемме об инверсии существует такой τ, что правый подтерм $x{:}\tau$, а левый подтерм (тоже x) имеет тип $\tau \to \sigma$.
По лемме о контекстах $x \in \mathrm{dom}(\Gamma)$ и должен иметь там *единственное* связывание по определению контекста. То есть $\tau = \tau \to \sigma$ — тип является подвыражением себя, чего не может быть, поскольку (*и пока*) типы конечны.

$$x{:}\tau \nvdash (x\,x):\sigma, \quad \nvdash \omega:\sigma, \quad \nvdash \mathbf{\Omega}:\sigma, \quad \nvdash \mathbf{Y}:\sigma.$$

Предтермы $\omega = \lambda x.\, x\,x$, $\mathbf{\Omega} = \omega\,\omega$ и $\mathbf{Y} = \lambda f\,.\,(\lambda x\,.\,f(x\,x))(\lambda x\,.\,f(x\,x))$ не имеют типа по свойству о типизируемости подтерма.

Свойства $\lambda\!\to$: лемма подстановки типа

Для $\sigma, \tau \in \mathbb{T}$ ***подстановку*** τ вместо α в σ обозначим $\sigma[\alpha := \tau]$.
Пример:

$$(\alpha \to \beta \to \alpha)[\alpha := \gamma \to \gamma] \;=\; (\gamma \to \gamma) \to \beta \to \gamma \to \gamma$$

Лемма подстановки типа

- $\Gamma \vdash M : \sigma \;\Rightarrow\; \Gamma[\alpha := \tau] \vdash M : \sigma[\alpha := \tau]$. ($\lambda\!\to$ а ля Карри)
- $\Gamma \vdash M : \sigma \;\Rightarrow\; \Gamma[\alpha := \tau] \vdash M[\alpha := \tau] : \sigma[\alpha := \tau]$. ($\lambda\!\to$ а ля Чёрч)

Доказательство. Индукциия по выводу $M : \sigma$.

Пример. Подстановка $[\alpha := \gamma \to \gamma]$:

$$x : \alpha \;\vdash\; (\lambda y^{\alpha} z^{\beta}.\, x) : \alpha \to \beta \to \alpha \quad \Rightarrow$$
$$x : \gamma \to \gamma \;\vdash\; (\lambda y^{\gamma \to \gamma} z^{\beta}.\, x) : (\gamma \to \gamma) \to \beta \to \gamma \to \gamma$$

Свойства $\lambda\to$: лемма подстановки терма

Лемма подстановки терма
Пусть $\Gamma, x:\sigma \vdash M:\tau$ и $\Gamma \vdash N:\sigma$, тогда $\Gamma \vdash M[x := N]:\tau$.
То есть, подходящая по типу ***подстановка сохраняет тип***.
Доказательство. Индукциия по генерации $\Gamma, x:\sigma \vdash M:\tau$.

Пример. Берём терм

$$x:\gamma\to\gamma \;\vdash\; (\lambda y^{\beta}.\, x):\beta\to\gamma\to\gamma$$

и подставляем в него вместо свободной переменной $x:\gamma\to\gamma$ терм $\mathbf{I}_{\gamma} \equiv \lambda p^{\gamma}.\, p$ подходящего типа $\gamma\to\gamma$. Получаем

$$\vdash\; (\lambda y^{\beta}\, p^{\gamma}.\, p):\beta\to\gamma\to\gamma$$

Свойства $\lambda\!\to$: редукция субъекта (1)

Теорема о редукции субъекта

Пусть $M \twoheadrightarrow_\beta N$. Тогда $\Gamma \vdash M\!:\!\sigma \;\Rightarrow\; \Gamma \vdash N\!:\!\sigma$.

То есть, β-***редукция терма сохраняет его тип***.

С «вычислительной» точки зрения это одно из ключевых свойств любой системы типов.

Следствие: множество типизируемых в $\lambda\!\to$ термов замкнуто относительно редукции.

Докажем теорему для $\lambda\!\to$ а ля Карри. Для $\lambda\!\to$ а ля Чёрч доказательство аналогичное, если задать редукцию в $\Lambda_{\mathbb{T}}$ как в Λ, на основе правила сокращения $(\lambda x\!:\!\tau.\, P)\, Q \to_\beta P[x := Q]$.

Свойства $\lambda{\rightarrow}$: редукция субъекта (2)

Теорема : $M \twoheadrightarrow_\beta N \;\wedge\; \Gamma \vdash M : \sigma \;\Rightarrow\; \Gamma \vdash N : \sigma$

Доказательство. Индукция по генерации $\twoheadrightarrow_\beta$.
Рассмотрим главный случай: $M \equiv (\lambda x.\, P)\, Q$ — редекс и $N \equiv P[x := Q]$ — его сокращение. Если

$$\Gamma \vdash (\lambda x.\, P)\, Q : \sigma,$$

то из леммы инверсии для некоторого τ имеем

$$\Gamma \vdash (\lambda x.\, P) : \tau \rightarrow \sigma \quad \text{и} \quad \Gamma \vdash Q : \tau.$$

Ещё раз используя лемму инверсии, получим

$$\Gamma, x : \tau \vdash P : \sigma \quad \text{и} \quad \Gamma \vdash Q : \tau$$

откуда по лемме подстановки терма

$$\Gamma \vdash P[x := Q] : \sigma. \quad \blacksquare$$

Свойства $\lambda\!\rightarrow$: экспансия субъекта (1)

Операция, обратная β-редукции называется экспансией (расширением).

Множество типизируемых в $\lambda\!\rightarrow$ термов не замкнуто относительно экспансии:

$$M \twoheadrightarrow_\beta N \;\wedge\; \Gamma \vdash N : \sigma \;\not\Rightarrow\; \Gamma \vdash M : \sigma$$

Действительно, рассмотрим терм $\mathbf{K\,I\,\Omega}$ ($\mathbf{I} \equiv \lambda x.\, x$, $\mathbf{K} \equiv \lambda x\, y.\, x$, $\mathbf{\Omega} \equiv \lambda x.\, x\, x$). Хотя $\mathbf{K\,I\,\Omega} \twoheadrightarrow_\beta \mathbf{I}$:

$$\mathbf{K\,I\,\Omega} \;\to_\beta\; (\lambda y.\, \mathbf{I})\,\mathbf{\Omega} \;\to_\beta\; \mathbf{I}$$

и $\vdash \mathbf{I} : \sigma \to \sigma$, но $\nvdash \mathbf{K\,I\,\Omega} : \sigma \to \sigma$, поскольку последнее содержит нетипизируемый подтерм.

Свойства $\lambda\!\to$: экспансия субъекта (2)

Для $\lambda\!\to$ а ля Чёрч экспансия не сохраняет тип только из-за возможного наличия нетипизируемого подтерма (чуть позже докажем).

Для $\lambda\!\to$ а ля Карри верно более сильное отрицание:

$$M \twoheadrightarrow_\beta N \;\wedge\; \Gamma \vdash M:\sigma \;\wedge\; \Gamma \vdash N:\tau \;\not\Rightarrow\; \Gamma \vdash M:\tau$$

Покажем это на примере.

Свойства $\lambda\!\rightarrow$: экспансия субъекта (3)

$$M \twoheadrightarrow_\beta N \ \wedge\ \Gamma \vdash M{:}\sigma \ \wedge\ \Gamma \vdash N{:}\tau \ \not\Rightarrow\ \Gamma \vdash M{:}\tau$$

Возьмём $M \equiv \mathbf{S}\,\mathbf{K}$ и $N \equiv \mathbf{K}_*$.

$$\begin{array}{rll}
\mathbf{S} \equiv \lambda f\,g\,z.\,f\,z\,(g\,z) & & \vdash \mathbf{S}{:}(\sigma\to\tau\to\rho)\to(\sigma\to\tau)\to\sigma\to\rho \\
\mathbf{K} \equiv \lambda x\,y.\,x & & \vdash \mathbf{K}{:}\sigma\to\tau\to\sigma \\
 & & \vdash (\mathbf{S}\,\mathbf{K}){:}(\sigma\to\tau)\to\sigma\to\sigma \\
\mathbf{K}_* \equiv \lambda x\,y.\,y & & \vdash \mathbf{K}_*{:}\tau\to\sigma\to\sigma
\end{array}$$

$$\begin{array}{lll}
\mathbf{S}\,\mathbf{K} & \to_\beta & \lambda g\,z.\,\mathbf{K}\,z\,(g\,z) \ \to_\beta \\
 & \to_\beta & \lambda g\,z.\,(\lambda y.\,z)\,(g\,z) \ \to_\beta\ \lambda g\,z.\,z \ \equiv_\alpha\ \mathbf{K}_*
\end{array}$$

В красной редукции потерялась информация о типе g, как о функциональном, $(g\,z){:}\tau \Rightarrow z{:}\sigma,\ g{:}\sigma\to\tau$.

Свойства $\lambda\!\to$: экспансия субъекта (4)

Для $\lambda\!\to$ в стиле Чёрча информация не теряется:

$$
\begin{array}{rcl}
\mathbf{S}_{\sigma\tau\rho} & \equiv & \lambda f^{\sigma\to\tau\to\rho}\, g^{\sigma\to\tau}\, z^{\sigma}.\, f\, z\, (g\, z) \\
\mathbf{K}_{\sigma\tau} & \equiv & \lambda x^{\sigma}\, y^{\tau}.\, x
\end{array}
$$

$$
\begin{array}{rcl}
\mathbf{S}_{\sigma\tau\sigma}\, \mathbf{K}_{\sigma\tau} & \to_\beta & \lambda g^{\sigma\to\tau}\, z^{\sigma}.\, \mathbf{K}_{\sigma\tau}\, z\, (g\, z) \ \to_\beta \\
& \to_\beta & \lambda g^{\sigma\to\tau}\, z^{\sigma}.\, (\lambda y^{\tau}.\, z)\, (g\, z) \ \to_\beta \ \lambda g^{\sigma\to\tau}\, z^{\sigma}.\, z \\
& \equiv_\alpha & \lambda x^{\sigma\to\tau}\, y^{\sigma}.\, y \ \equiv \ \mathbf{K}_{*(\sigma\to\tau)\sigma}
\end{array}
$$

Можно переименовать связанные переменные, но не их типы!

Свойства $\lambda\!\to$: единственность типа

Теорема о единственности типа для $\lambda\!\to$ а ля Чёрч

- Пусть $\Gamma \vdash M : \sigma$ и $\Gamma \vdash M : \tau$. Тогда $\sigma \equiv \tau$. Терм в $\lambda\!\to$ имеет единственный тип.
- Пусть $\Gamma \vdash M : \sigma$, $\Gamma \vdash N : \tau$ и $M =_\beta N$. Тогда $\sigma \equiv \tau$. Типизируемые β-конвертируемые термы имеют один тип.

Доказательство.
(1) Индукция по структуре M.
(2) По теореме Чёрча-Россера, теореме редукции субъекта с использованием (1). ∎

Для систем в стиле Карри единственности типа нет.

Связь между системами Карри и Чёрча

Можно задать стирающее отображение $|\cdot| : \Lambda_{\mathbb{T}} \to \Lambda$:

$$\begin{array}{rcl} |x| & \equiv & x \\ |M\,N| & \equiv & |M|\,|N| \\ |\lambda x\!:\!\sigma.\,M| & \equiv & \lambda x.\,|M| \end{array}$$

Все атрибутированные типами термы из версии Чёрча $\lambda\!\to$ «проектируются» в термы в версии Карри:

- $M \in \Lambda_{\mathbb{T}} \;\wedge\; \Gamma \vdash_{\text{Ч}} M\!:\!\sigma \;\Rightarrow\; \Gamma \vdash_{\text{К}} |M|\!:\!\sigma$

Термы из версии Карри $\lambda\!\to$ могут быть «подняты» в термы из версии Чёрча:

- $M \in \Lambda \;\wedge\; \Gamma \vdash_{\text{К}} M\!:\!\sigma \;\Rightarrow\; \exists N \in \Lambda_{\mathbb{T}}\,[\Gamma \vdash_{\text{Ч}} N\!:\!\sigma \;\wedge\; |N| \equiv M]$

Для произвольного типа $\sigma \in \mathbb{T}$ выполняется

$$\sigma \text{ населено в } \lambda\!\to\text{-Карри} \;\Leftrightarrow\; \sigma \text{ населено в } \lambda\!\to\text{-Чёрч}$$

Главный тип (principle type)

В версии Чёрча $\lambda\!\to$ термы атрибутированны типами, поэтому тип терма единственен. Для $\mathbf{S}_{\sigma\tau\rho} \equiv \lambda f^{\sigma\to\tau\to\rho}\, g^{\sigma\to\tau}\, z^{\sigma}.\, f\, z\, (g\, z)$:

$$\mathbf{S}_{\sigma\tau\rho} : (\sigma\to\tau\to\rho)\to(\sigma\to\tau)\to\sigma\to\rho$$

$$\mathbf{S}_{\sigma\tau\sigma} : (\sigma\to\tau\to\sigma)\to(\sigma\to\tau)\to\sigma\to\sigma$$

$$\mathbf{S}_{(\tau\to\rho)\tau\rho} : ((\tau\to\rho)\to\tau\to\rho)\to((\tau\to\rho)\to\tau)\to(\tau\to\rho)\to\rho$$

Любой из этих типов можно присвоить терму $\mathbf{S} \equiv \lambda f\, g\, z.\, f\, z\, (g\, z)$ в версии Карри.

Однако, первый «лучше» в том смысле, что остальные получаются из него подстановкой типа вместо типовой переменной.

Вывод главного типа (пример)

$$\lambda x\, y.\, y\,(\lambda z.\, y\, x) \qquad\qquad \lambda x^{\alpha} y^{\beta}.\, \underbrace{y^{\beta}\,(\lambda z^{\gamma}.\, \overbrace{y^{\beta}\, x^{\alpha}}^{\delta})}_{\varepsilon}$$

1. Присвоим типовую переменную всем термовым переменным: $x^{\alpha}, y^{\beta}, z^{\gamma}$.
2. Присвоим типовую переменную всем *аппликативным* подтермам: $(y\,x):\delta$, $(y\,(\lambda z.\, y\, x)):\varepsilon$.
3. Выпишем уравнения (ограничения) на типы, необходимые для типизируемости терма: $\beta \sim \alpha \to \delta, \quad \beta \sim (\gamma \to \delta) \to \varepsilon$.
4. Найдём *главный унификатор* для типовых переменных (подстановку), дающий решения уравнений:
$\alpha := \gamma \to \delta, \quad \beta := (\gamma \to \delta) \to \varepsilon, \quad \delta := \varepsilon$.
5. Главный тип $(\lambda x\, y.\, y\,(\lambda z.\, y\, x)):(\gamma \to \varepsilon) \to ((\gamma \to \varepsilon) \to \varepsilon) \to \varepsilon$.

Подстановка типов: расширенное определение

Подстановка типов — это операция $S{:}\mathbb{T} \to \mathbb{T}$, такая что

$$S(\sigma \to \tau) \;\equiv\; S(\sigma) \to S(\tau)$$

Обычно подстановка тождественна на всех типовых переменных, кроме конечного носителя $\mathrm{sup}(S) = \{\alpha \mid S(\alpha) \not\equiv \alpha\}$.

Пример подстановки $S = [\alpha := \gamma \to \beta, \beta := \alpha \to \gamma]$.

Тождественную подстановку (с пустым носителем) обозначим $[\,]$.

Подстановка выполняется *параллельно*; для $\tau = \alpha \to \beta \to \gamma$

$$\begin{aligned} S(\tau) &= [\alpha := \gamma \to \beta, \beta := \alpha \to \gamma](\alpha \to \beta \to \gamma) \\ &= (\gamma \to \beta) \to (\alpha \to \gamma) \to \gamma \end{aligned}$$

Композиция подстановок

Композиция подстановок — подстановка с носителем, являющимся объединением носителей. Для

$$
\begin{aligned}
S &= [\alpha := \gamma \to \beta, \beta := \alpha \to \gamma]; \\
\mathsf{T} &= [\alpha := \beta \to \gamma, \gamma := \beta] \text{ имеем} \\
\mathsf{T} \circ S &= [\alpha := \mathsf{T}(S(\alpha)), \beta := \mathsf{T}(S(\beta)), \gamma := \mathsf{T}(S(\gamma))], \text{ то есть} \\
\mathsf{T} \circ S &= [\alpha := \beta \to \beta, \beta := (\beta \to \gamma) \to \beta, \gamma := \beta]
\end{aligned}
$$

Подстановки — моноид относительно $\circ$ с единицей $[]$.

Сравним, например, $(\mathsf{T} \circ S)(\tau)$ и $\mathsf{T}(S(\tau))$ для $\tau = \alpha \to \beta \to \gamma$

$$
\begin{aligned}
(\mathsf{T} \circ S)(\tau) &= (\beta \to \beta) \to ((\beta \to \gamma) \to \beta) \to \beta; \\
\mathsf{T}(S(\tau)) &= \mathsf{T}((\gamma \to \beta) \to (\alpha \to \gamma) \to \gamma) \\
&= (\beta \to \beta) \to ((\beta \to \gamma) \to \beta) \to \beta
\end{aligned}
$$

Унификатор

Унификатор для типов σ и τ — это подстановка S, такая что $S(\sigma) \equiv S(\tau)$.

Пример. Пусть $\sigma = \beta \to \alpha \to \beta$ и $\tau = (\gamma \to \gamma) \to \delta$. Их унификатор

$$\begin{aligned} S &= [\beta := \gamma \to \gamma, \delta := \alpha \to \gamma \to \gamma] \\ S(\sigma) \equiv S(\tau) &= (\gamma \to \gamma) \to \alpha \to \gamma \to \gamma \end{aligned}$$

Унификатор S — это ***главный унификатор*** для σ и τ, если для любого другого унификатора S' существует подстановка T, такая что

$$S' \equiv T \circ S$$

Пример.

$$\begin{aligned} S' &= [\beta := \gamma \to \gamma, \alpha := \varepsilon \to \varepsilon, \delta := \varepsilon \to \varepsilon \to (\gamma \to \gamma)] \\ S' &= [\alpha := \varepsilon \to \varepsilon] \circ S \end{aligned}$$

Теорема унификации

Теорема унификации. Существует алгоритм унификации $\mathcal{U}$, который для заданных типов σ и τ возвращает:

- главный унификатор S для σ и τ, если σ и τ могут быть унифицированы;
- сообщение об ошибке в противном случае.

Алгоритм $\mathcal{U}(\sigma, \tau)$ позволяет искать «минимальное» решение уравнения на типы $\sigma \sim \tau$.

Ключевой момент всех рассуждений про унификацию:

$$\sigma_1 \to \sigma_2 \equiv \tau_1 \to \tau_2 \ \Leftrightarrow\ \sigma_1 \equiv \tau_1 \ \wedge\ \sigma_2 \equiv \tau_2$$

Вывод типов : алгоритм унификации $\mathcal{U}$

$$
\begin{array}{lcl}
\mathcal{U}(\alpha, \alpha) & = & [\,] \\
\mathcal{U}(\alpha, \tau) \mid \alpha \in FV(\tau) & = & \text{ошибка} \\
\mathcal{U}(\alpha, \tau) \mid \alpha \notin FV(\tau) & = & [\alpha := \tau] \\
\mathcal{U}(\sigma_1 \to \sigma_2, \alpha) & = & \mathcal{U}(\alpha, \sigma_1 \to \sigma_2) \\
\mathcal{U}(\sigma_1 \to \sigma_2, \tau_1 \to \tau_2) & = & \mathcal{U}\,(\mathcal{U}(\sigma_2, \tau_2)\sigma_1, \mathcal{U}(\sigma_2, \tau_2)\tau_1) \circ \mathcal{U}(\sigma_2, \tau_2)
\end{array}
$$

- $\mathcal{U}(\sigma, \tau)$ завершается, поскольку на каждом шаге сокращает либо количество $\to$, либо количество типовых переменных.
- $\mathcal{U}(\sigma, \tau)$ унифицирует. По индукции; используем, что если S унифицирует (σ, τ), то $S \circ [\alpha := \rho]$ унифицирует $(\sigma \to \alpha, \tau \to \rho)$.
- $\mathcal{U}(\sigma, \tau)$ даёт главный унификатор. По индукции; см. [TAPL 22.4]

Унификация системы уравнений на типы

Если есть система уравнений $E = \{\sigma_1 \sim \tau_1, \ldots, \sigma_n \sim \tau_n\}$, то её унификатором называют такую подстановку S, что

$$S(\sigma_1) \equiv S(\tau_1) \ \wedge \ \ldots \ \wedge \ S(\sigma_n) \equiv S(\tau_n)$$

При этом пишут $S \models E$ (подстановка S унифицирует систему E).

Главный унификатор для системы E ищется с помощью алгоритма U следующим образом:

$$U(E) = U(\sigma_1 \to \ldots \to \sigma_n, \tau_1 \to \ldots \to \tau_n)$$

Алгоритм унификации $\mathcal{U}$: пример

$$
\begin{array}{lcl}
\mathcal{U}(\alpha, \alpha) & = & [\,] \\
\mathcal{U}(\alpha, \tau) \mid \alpha \in FV(\tau) & = & \text{ошибка} \\
\mathcal{U}(\alpha, \tau) \mid \alpha \notin FV(\tau) & = & [\alpha := \tau] \\
\mathcal{U}(\sigma_1 \to \sigma_2, \alpha) & = & \mathcal{U}(\alpha, \sigma_1 \to \sigma_2) \\
\mathcal{U}(\sigma_1 \to \sigma_2, \tau_1 \to \tau_2) & = & \mathcal{U}\left(\mathcal{U}(\sigma_2, \tau_2)\sigma_1, \mathcal{U}(\sigma_2, \tau_2)\tau_1\right) \circ \mathcal{U}(\sigma_2, \tau_2)
\end{array}
$$

Для $\lambda x\, y.\, y\,(\lambda z.\, y\, x)$ система уравнений на типы имела вид $E = \{\beta \sim (\gamma \to \delta) \to \varepsilon, \beta \sim \alpha \to \delta\}$. Алгоритм $\mathcal{U}$ даёт:

$$
\begin{array}{lcl}
\mathcal{U}(E) & = & \mathcal{U}(\beta \to \beta, ((\gamma \to \delta) \to \varepsilon) \to (\alpha \to \delta)) \\
& = & \mathcal{U}(\mathcal{U}(\beta, \alpha \to \delta)\beta, \mathcal{U}(\beta, \alpha \to \delta)(\gamma \to \delta) \to \varepsilon) \circ \mathcal{U}(\beta, \alpha \to \delta) \\
& = & \mathcal{U}(\alpha \to \delta, (\gamma \to \delta) \to \varepsilon) \circ [\beta := \alpha \to \delta] \\
& = & [\alpha := \gamma \to \varepsilon] \circ [\delta := \varepsilon] \circ [\beta := \alpha \to \delta] \\
& = & [\alpha := \gamma \to \varepsilon, \delta := \varepsilon, \beta := (\gamma \to \varepsilon) \to \varepsilon]
\end{array}
$$

Домашнее задание (1)

Проследите за изменениями в работе алгоритма $\mathcal{U}$, при перестановке элементов в E: $\mathrm{E} = \{\beta \sim \alpha \to \delta, \beta \sim (\gamma \to \delta) \to \varepsilon\}$ Что изменится?

Реализуйте алгоритм $\mathcal{U}$ на каком-нибудь ЯП.

Алгоритм построения ограничений

Теорема. Для любых контекста Γ, терма $M \in \Lambda$ ($FV(M) \subseteq \mathrm{dom}(\Gamma)$) и типа $\sigma \in \mathbb{T}$ существует такое множество уравнений на типы $E = E(\Gamma, M, \sigma)$, что для любой подстановки S:

- $S \vDash E(\Gamma, M, \sigma) \;\Rightarrow\; S(\Gamma) \vdash M{:}S(\sigma)$;
- $S(\Gamma) \vdash M{:}S(\sigma) \;\Rightarrow\; S' \vDash E(\Gamma, M, \sigma)$, для некоторой S', имеющего тот же эффект, что и S, на типовых переменных в Γ и σ.

Алгоритм E можно задать так

$$
\begin{array}{rcl}
E(\Gamma, x, \sigma) & = & \{\sigma \sim \Gamma(x)\} \\
E(\Gamma, M\,N, \sigma) & = & E(\Gamma, M, \alpha \to \sigma) \cup E(\Gamma, N, \alpha) \\
E(\Gamma, \lambda x.\, M, \sigma) & = & E(\Gamma \cup \{x{:}\alpha\}, M, \beta) \cup \{\alpha \to \beta \sim \sigma\}
\end{array}
$$

Здесь α и β — всякий раз «свежие»!

Главная пара (Principle Pair) и главный тип (Principle Type)

Для $M \in \Lambda$ ***главной парой*** называют пару (Γ, σ), такую что

- $\Gamma \vdash M : \sigma$
- $\Gamma' \vdash M : \sigma' \;\Rightarrow\; \exists S \;\left[S(\Gamma) \subseteq \Gamma' \;\wedge\; S(\sigma) \equiv \sigma'\right]$

Если $(\Gamma, \sigma) = PP(M)$, то $FV(M) = \mathrm{dom}(\Gamma)$.

Пример. Для $M = \lambda x.\, x\, y$ имеем $PP(M) = (y : \alpha, (\alpha \to \beta) \to \beta)$

$$y : \alpha \vdash (\lambda x.\, x\, y) : (\alpha \to \beta) \to \beta$$

Для $M \in \Lambda^0$ ***главным типом*** называют тип σ, такой что

- $\vdash M : \sigma$
- $\vdash M : \sigma' \;\Rightarrow\; \exists S \;\left[S(\sigma) \equiv \sigma'\right]$

Теорема Хиндли – Милнера

Существует алгоритм PP, возвращающий для $M \in \Lambda$

- главную пару (Γ, σ), если M имеет тип;
- сообщение об ошибке в противном случае.

Пусть $FV(M) = \{x_1, \ldots, x_n\}$, $\Gamma_0 = \{x_1 : \alpha_1, \ldots, x_n : \alpha_n\}$ и $\sigma_0 = \beta$.

Алгоритм PP можно задать так

$$
\begin{array}{lcl}
PP(M) \mid U(E(\Gamma_0, M, \sigma_0)) = \text{ошибка} & = & \text{ошибка} \\
PP(M) \mid U(E(\Gamma_0, M, \sigma_0)) = S & = & (S(\Gamma_0), S(\sigma_0))
\end{array}
$$

Теорема Хиндли – Милнера (следствие)

Существует алгоритм PT, возвращающий для $M \in \Lambda^0$

- главный тип σ, если M имеет тип;
- сообщение об ошибке в противном случае.

Доказательство. Пусть M замкнут и $PP(M) = (\Gamma, \sigma)$. Но $\Gamma = \varnothing$ и мы можем положить $PT(M) = \sigma$. ■

Домашнее задание (2)

(забыл дать в прошлый раз)
Реализуйте алгоритм редукции терма к нормальной форме с помощью нормальной стратегии на каком-нибудь ЯП. Сделайте также пошаговую версию.

Докажите $\nvdash \mathbf{\Omega_3}{:}\sigma$. ($\mathbf{\Omega_3} \equiv \omega_3\,\omega_3 \equiv (\lambda x.\, x\,x\,x)\,(\lambda x.\, x\,x\,x)$)

Найдите наиболее общий тип (или докажите нетипизируемость) для

- $\lambda z\,x.\, z\,(x\,(\lambda y.\, y\,z))$
- $\lambda z\,x.\, z\,(x\,(\lambda y.\, y\,x))$
- $\lambda x\,y.\, x\,(y\,(\lambda z.\, x\,z\,z))\,(y\,(\lambda z.\, x\,z\,z))$

Реализуйте алгоритмы РР и РТ на каком-нибудь ЯП.

Литература (1)

LCWT гл. 3, 4.4
Henk Barendregt, Lambda calculi with types,
Handbook of logic in computer science (vol. 2), Oxford University Press, 1993

ITT гл. 4
Herman Geuvers, Introduction to Type Theory
Alfa Lernet Summer school 2008, Uruguay
http://www.cs.ru.nl/H.Geuvers/Uruguay2008SummerSchool.html/

Литература (2)

TAPL гл. 9, 22
Benjamin C. Pierce, Types and Programming Languages, MIT Press, 2002
http://www.cis.upenn.edu/ bcpierce/tapl

I2FP гл. 4.1
John Harrison, Introduction to Functional Programming
http://www.cl.cam.ac.uk/teaching/Lectures/funprog-jrh-1996/
русский перевод: http://code.google.com/p/funprog-ru/