

Генерация зависимых языков по спецификации пользователя

Выступающий: Шамиль Гарифуллин
Руководитель: Валерий Исаев

Санкт-Петербургский Академический Университет

СПбАУ, 2017

Соответствие Карри-Говарда

Описаны в [PV07; Isa16]

Сигнатура – пара $(\mathcal{S}, \mathcal{F})$, \mathcal{S} – произвольное множество, \mathcal{F} – множество и функция, которая каждому $f \in \mathcal{F}$ сопоставляет его тип

$$\overline{x : s}, x \in Var, x : s$$

$$\frac{t_1 : s_1 \quad \dots \quad t_n : s_n}{f(t_1, \dots, t_n) : s}, f : s_1 \times \dots \times s_n \rightarrow s$$

Пример определения языка – теории

В ТТ

Формула в сигнатуре Σ – это выражение вида $t = t'$, где t и t' – термы сигнатуры Σ одного сорта.

$$\varphi \vdash^V \varphi \text{ (b1)} \quad \frac{\varphi \vdash^V \psi \quad \psi \vdash^V \chi}{\varphi \vdash^V \chi} \text{ (b2)} \quad \varphi \vdash^V \top \text{ (b3)}$$

$$\varphi \wedge \psi \vdash^V \varphi \text{ (b4)} \quad \varphi \wedge \psi \vdash^V \psi \text{ (b5)} \quad \frac{\varphi \vdash^V \psi \quad \varphi \vdash^V \chi}{\varphi \vdash^V \psi \wedge \chi} \text{ (b6)}$$

$$\vdash^x x \downarrow \text{ (a1)} \quad x = y \wedge \varphi \vdash^{V, x, y} \varphi[y/x] \text{ (a2)}$$

$$\frac{\varphi \vdash^V \psi}{\varphi[t/x] \vdash^{V, V'} \psi[t/x]}, \quad x \in FV(\varphi), \quad t \in T(V') \text{ (a3)}$$

Зависимые языки – веселый пример

из ТТ

Формула в сигнатуре Σ – это выражение вида $t = t'$, где t и t' – термы сигнатуры Σ одного сорта.

$$\varphi \vdash^V \varphi \text{ (b1)} \quad \frac{\varphi \vdash^V \psi \quad \psi \vdash^V \chi}{\varphi \vdash^V \chi} \text{ (b2)} \quad \varphi \vdash^V \top \text{ (b3)}$$

$$\varphi \wedge \psi \vdash^V \varphi \text{ (b4)} \quad \varphi \wedge \psi \vdash^V \psi \text{ (b5)} \quad \frac{\varphi \vdash^V \psi \quad \varphi \vdash^V \chi}{\varphi \vdash^V \psi \wedge \chi} \text{ (b6)}$$

$$\vdash^x x \downarrow \text{ (a1)} \quad x = y \wedge \varphi \vdash^{V, x, y} \varphi[y/x] \text{ (a2)}$$

$$\frac{\varphi \vdash^V \psi}{\varphi[t/x] \vdash^{V, V'} \psi[t/x]}, \quad x \in FV(\varphi), \quad t \in T(V') \text{ (a3)}$$

- Создать язык для генерации проверки типов в зависимых языках
- Сделать его удобным для использования
 - Вывод ошибок пользователя при написании доказательств
 - Простота добавления новых конструкций
 - Автоматизация частей доказательств
- Написать пример работы для ознакомления с использованием EDSL

Вот пример спецификации языка

- Проверка типов термов, формул и секвенций
- Вывод ошибок пользователя на уровне типов
- `typeclass` для (S, \mathcal{F}) и для аксиом пользователя
- Проверка доказательств с выводом ошибок применения правил

Проверка нашего языка?

- Неудобство использования правил
 - Неудобны для применения на концептуальном уровне
 - Невозможность расширять пользовательскими правилами
- Вывод термов и доказательств на экран
- $t(x, y) ::= f(g(x, f(x, y)), g(y, x))$

Генерация языка - смотрим и все рекурсивно определено

$$\begin{array}{c}
 \frac{}{\Gamma \vdash x \downarrow} \qquad \frac{\Gamma \vdash a = b}{\Gamma \vdash b = a} \\
 \\
 \frac{}{\varphi_1, \dots, \varphi_n \vdash \varphi_i} \qquad \frac{\Gamma \vdash a = b \quad \Gamma \vdash \varphi[a/x]}{\Gamma \vdash \varphi[b/x]} \\
 \\
 \frac{\Gamma \vdash R(t_1, \dots, t_n)}{\Gamma \vdash t_i \downarrow} \qquad \frac{\Gamma \vdash \sigma(t_1, \dots, t_n) \downarrow}{\Gamma \vdash t_i \downarrow}
 \end{array}$$

$$\frac{\Gamma \vdash \varphi_1[t_1/x_1, \dots, t_k/x_k] \quad \dots \quad \Gamma \vdash \varphi_n[t_1/x_1, \dots, t_k/x_k]}{\Gamma \vdash \varphi[t_1/x_1, \dots, t_k/x_k]}$$

для всех аксиом $\varphi_1, \dots, \varphi_n \vdash \frac{x_1:s_1, \dots, x_k:s_k}{\varphi}$ и термов t_i сорта s_i

Индексы де брейна - опиши проблемы подстановки и альфа равенства, опиши классность на уровне типов, опиши контексты

- Реализованы новые правила вывода
- Возможность добавлять пользовательские правила [Swi08; Bah14]
- `type MyProof = (Rule :+: Trans :+: AxiomRule) Axioms Sort Fun`
- Общий Pretty Printing расширяемый пользователем

- Высокоуровневый язык спецификации языков программирования с зависимыми типами
- Генерация модуля с функциями проверки типов и приведения в нормальную форму
 - Индексы де Брейна на уровне типов данных
 - Проверка типов через нормализацию
- Примеры спецификации языка
- ссылка на репо!

Дальнейшее работа??

- Проверка программ - отличие в функциях и парсере

Список литературы



Erik Palmgren и Steven J Vickers. “Partial Horn logic and cartesian categories”. В: *Annals of Pure and Applied Logic* 145.3 (2007), с. 314—353.



Wouter Swierstra. “Data types à la carte”. В: *Journal of functional programming* 18.04 (2008), с. 423—436.



Patrick Bahr. “Composing and decomposing data types: a closed type families implementation of data types à la carte”. В: *Proceedings of the 10th ACM SIGPLAN workshop on Generic programming*. ACM. 2014, с. 71—82.



Valery Isaev. “Algebraic Presentations of Dependent Type Theories”. В: *arXiv preprint arXiv:1602.08504* (2016).