모듈이 있는 언어에서 타입 안전성 증명

Logical Relations and Subtyping

이준협

2024년 1월 12일

ROPAS@SNU

동기: 타입의 안전성 증명

- 따로분석을 타입분석의 경우로 확장하고 싶다.
- 그 전, <u>언어에 맞는 타입규칙</u>을 만들고, <u>안전성 증명</u>을 해야한다.

문제: 의미구조와 아래타입

1. 의미구조가 초기 상태와 최종 결과를 바로 연관시킨다. (큰 보폭)
   → 직관적이지만…
   → Progress & Preservation 스타일의 증명 어려움
2. 풍부한 타입규칙을 위해 아래타입 (subtype) 도입
   → 예시: `x`만 쓰는 함수에게 `x,y`를 내보내는 모듈이 주어져도 안전
   → 타입 안전성 증명을 어떻게 해야할까?

해답: Logical Relations

- Simply Typed λ의 예시: $\tau, A, B \in \text{Type}, e \in \text{Expr}, v \in \text{Val}$
  $\boxed{R_\tau \subseteq \text{Val}^n}$는 다음과 같을 때 n-**ary logical relation**이다:

  모든 $(v_1, \dots, v_n) \in R_A$에 대해
  어떤 $(v_1', \dots, v_n') \in R_B$가 존재해 $e_i[v_i/x_i] \Downarrow v_i'$ **이면이**
  $(\lambda x_1.e_1, \dots, \lambda x_n.e_n) \in R_{A \to B}$ **이다.**
- "연관된" 입력을 "연관된" 출력으로 보내는 함수들의 집합.
- $R \in \text{Type} \to \mathscr{P}(\text{Val}^n)$: 타입의 "구체화"로 이해

타입 안전성

- $n = 1$인 경우(unary) 사용.
  목표: $\boxed{\vdash e : \tau \Rightarrow \exists v \text{ s.t } e \Downarrow v \text{ and } v \in R_\tau}$
- **겉모습**만 보고 추론된 타입이 실제로 프로그램의 **의미**를 포섭한다

모듈이 있는 언어: 겉모습

$$
\begin{array}{lcll}
\text{Identifiers} & x & \in & \text{Var} & \\
\text{Expression} & e & \to & x \mid \lambda x.e \mid e\ e & \lambda\text{-calculus} \\
 & & \mid & e \rtimes e & \text{linked expression} \\
 & & \mid & \varepsilon & \text{empty module} \\
 & & \mid & x = e\ ;\ e & \text{binding}
\end{array}
$$

의미공간

$$
\begin{array}{rcll}
\text{Environment} & \sigma & \in & \text{Env} \\
\text{Value} & v & \in & \text{Val} \triangleq \text{Env} + \text{Var} \times \text{Expr} \times \text{Env} \\
\text{Environment} & \sigma & \rightarrow & \bullet & \text{empty stack} \\
 & & \mid & (x, v) :: \sigma & \text{value binding} \\
\text{Value} & v & \rightarrow & \sigma & \text{exported environment} \\
 & & \mid & \langle \lambda x.e, \sigma \rangle & \text{closure}
\end{array}
$$

큰 보폭 실행의미

$$\boxed{(e, \sigma) \Downarrow v}$$

$$\textsc{Id}\ \frac{v = \sigma(x)}{(x, \sigma) \Downarrow v} \qquad \textsc{Fn}\ \frac{}{(\lambda x.e, \sigma) \Downarrow \langle \lambda x.e, \sigma \rangle}$$

$$\textsc{App}\ \frac{(e_1, \sigma) \Downarrow \langle \lambda x.e_\lambda, \sigma_\lambda \rangle \qquad (e_2, \sigma) \Downarrow v \qquad (e_\lambda, (x, v) :: \sigma_\lambda) \Downarrow v'}{(e_1\ e_2, \sigma) \Downarrow v'}$$

$$\textsc{Link}\ \frac{(e_1, \sigma) \Downarrow \sigma' \qquad (e_2, \sigma') \Downarrow v}{(e_1 \rtimes e_2, \sigma) \Downarrow v} \qquad \textsc{Empty}\ \frac{}{(\varepsilon, \sigma) \Downarrow \bullet} \qquad \textsc{Bind}\ \frac{(e_1, \sigma) \Downarrow v \qquad (e_2, (x, v) :: \sigma) \Downarrow \sigma'}{(x = e_1; e_2, \sigma) \Downarrow (x, v) :: \sigma'}$$

타입

$$
\begin{array}{rccll}
\text{Types} & \tau & \rightarrow & \Gamma & \text{module type} \\
 & & \mid & \tau \rightarrow \tau & \text{function type} \\
\text{Type Environment} & \Gamma & \rightarrow & \bullet & \text{empty environment} \\
 & & \mid & (x, \tau) :: \Gamma & \text{type binding}
\end{array}
$$

타입규칙

$$\boxed{\Gamma \vdash e : \tau}$$

$$\textsc{T-Id}\ \frac{\tau = \Gamma(x)}{\Gamma \vdash x : \tau} \qquad \textsc{T-Fn}\ \frac{(x, \tau_1) :: \Gamma \vdash e : \tau_2}{\Gamma \vdash \lambda x.e : \tau_1 \to \tau_2} \qquad \textsc{T-App}\ \frac{\Gamma \vdash e_1 : \tau_1 \to \tau \qquad \Gamma \vdash e_2 : \tau_2 \qquad \tau_1 \geq \tau_2}{\Gamma \vdash e_1\ e_2 : \tau}$$

$$\textsc{T-Link}\ \frac{\Gamma \vdash e_1 : \Gamma_1 \qquad \Gamma_1 \vdash e_2 : \tau_2}{\Gamma \vdash e_1 \rtimes e_2 : \tau_2} \qquad \textsc{T-Mt}\ \frac{}{\Gamma \vdash \varepsilon : \bullet} \qquad \textsc{T-Bind}\ \frac{\Gamma \vdash e_1 : \tau_1 \qquad (x, \tau_1) :: \Gamma \vdash e_2 : \Gamma_2}{\Gamma \vdash x = e_1; e_2 : (x, \tau_1) :: \Gamma_2}$$

아래타입

$$\boxed{\tau \geq \tau}$$

$$\frac{}{\bullet \geq \bullet}\ \textsc{Nil}$$

$$\frac{x \notin \mathsf{dom}(\Gamma) \qquad \Gamma \geq \Gamma'}{\Gamma \geq (x, \tau) :: \Gamma'}\ \textsc{ConsFree}$$

$$\frac{\Gamma(x) \geq \tau \qquad \Gamma - x \geq \Gamma'}{\Gamma \geq (x, \tau) :: \Gamma'}\ \textsc{ConsBound}$$

$$\frac{\tau_2 \geq \tau_1 \qquad \tau_1' \geq \tau_2'}{\tau_1 \to \tau_1' \geq \tau_2 \to \tau_2'}\ \textsc{Arrow}$$

- $\tau_1 \geq \tau_2$: τ_1 자리에 τ_2를 넣어도 안전하다.
- Reflexivity, Transitivity 만족함을 증명할 수 있음.

(Unary) Logical Relation

Value Relation $\boxed{V[\![\tau]\!]}$

$$
\begin{array}{rcl}
V[\![\bullet]\!] & \triangleq & \text{Env} \\
V[\![(x, \tau) :: \Gamma]\!] & \triangleq & \{\sigma | \sigma(x) \in V[\![\tau]\!] \land \sigma - x \in V[\![\Gamma - x]\!]\} \\
V[\![\tau_1 \to \tau_2]\!] & \triangleq & \{\langle \lambda x.e, \sigma \rangle | \forall v \in V[\![\tau_1]\!] : (e, (x, v) :: \sigma) \in E[\![\tau_2]\!]\}
\end{array}
$$

Expression Relation $\boxed{E[\![\tau]\!]}$

$$
E[\![\tau]\!] \quad \triangleq \quad \{(e, \sigma) | \exists v \in V[\![\tau]\!] : (e, \sigma) \Downarrow v\}
$$

- 타입에 대한 귀납법으로 정의됨(well-founded)
- $\bullet$에 대한 해석: $V[\![\bullet]\!] = \text{Env}$ vs $V[\![\bullet]\!] = \{\bullet\}$

타입 안전성

Semantic Typing $\boxed{\Gamma \vDash e : \tau}$

$$\Gamma \vDash e : \tau \quad \triangleq \quad \forall \sigma \in V[\![\Gamma]\!] : (e, \sigma) \in E[\![\tau]\!]$$

Theorem (Type Safety)

$$\Gamma \vdash e : \tau \Rightarrow \Gamma \vDash e : \tau$$

- 타입규칙에 대한 귀납법으로 증명
  $+$ 보조정리 2개

보조정리: 타입규칙이 잘 정의됨

- $\vdash$를 정의하는 규칙들은 $\models$의 경우에도 성립한다
- 즉, $\vdash$는 잘 정의되었다.

예시:

$$\textsc{T-Id}\ \frac{\tau = \Gamma(x)}{\Gamma \models x : \tau} \qquad \textsc{T-Fn}\ \frac{(x, \tau_1) :: \Gamma \models e : \tau_2}{\Gamma \models \lambda x.e : \tau_1 \to \tau_2} \qquad \textsc{T-App}\ \frac{\Gamma \models e_1 : \tau_1 \to \tau \qquad \Gamma \models e_2 : \tau_2 \qquad \tau_1 \geq \tau_2}{\Gamma \models e_1\ e_2 : \tau}$$

보조정리: 아래타입이 잘 정의됨

Lemma (Subtyping is well-defined)

$$\tau_1 \geq \tau_2 \Rightarrow V[\![\tau_1]\!] \supseteq V[\![\tau_2]\!]$$

- 성립하는 이유: $V[\![\bullet]\!] = \mathsf{Env}$로 정의했기 때문
- 모든 Γ에 대해, $\bullet \geq \Gamma$임.

만약 $V[\![\bullet]\!] = \{\bullet\}$ 이었다면?

- 아래타입 관계를 해석할 수 없게 됨
- $\sigma \in V[\![\Gamma]\!]$ 의 의미가 강력해짐:
  Γ 이상을 내보내는→정확히 Γ 만큼 내보내는
- 예시: $\sigma_1 \in V[\![\Gamma_1]\!] \wedge \sigma_2 \in V[\![\Gamma_2]\!] \Rightarrow \sigma_1 +\!\!+ \sigma_2 \in V[\![\Gamma_1 +\!\!+^{\#} \Gamma_2]\!]$ 인 $+\!\!+^{\#}$?
  1. $V[\![\bullet]\!] = \mathsf{Env}$ 일 때: $\Gamma_1 +\!\!+^{\#} \Gamma_2 \triangleq \Gamma_1$
  2. $V[\![\bullet]\!] = \{\bullet\}$ 일 때: $\Gamma_1 +\!\!+^{\#} \Gamma_2 \triangleq \Gamma_1 +\!\!+ \Gamma_2$

재귀적 모듈을 지원하려면?

$$\frac{\substack{\textsc{T-Bind} \\ (x, \tau_1) :: \Gamma \vdash e_1 : \tau_1 \qquad (x, \tau_1) :: \Gamma \vdash e_2 : \Gamma_2}}{\Gamma \vdash x = e_1; e_2 : (x, \tau_1) :: \Gamma_2}$$

이면?

$$\frac{\substack{\textsc{T-Bind} \\ (x, \tau) :: \Gamma \vdash x : \tau \qquad (x, \tau) :: \Gamma \vdash \varepsilon : \bullet}}{\Gamma \vdash x = x; \varepsilon : (x, \tau) :: \bullet}$$

이 모든 Γ와 τ에 대해 성립

- “어떤 위치”에서 재귀적 정의된 값이 쓰이는지 중요
  - “A practical mode system for recursive definitions”, POPL 2021