프로그램 따로분석의 이론적 기틀

이준협

2024년 4월 19일

ROPAS S&T

따로분석?

- 따로분석이란,
- 모든 것이 알려지기 전에,
- 프로그램의 의미를 포섭하는 분석이다.

왜 필요한가?

```
int *x;
void f(void) {
  if (*x == *x) g(); /* call to unknown function */
  *x = 42; /* what happens here? */
}
```

- 코드 전체가 주어지지 않는 일이 빈번하다.
  ex. 외부 라이브러리 함수를 부르는 경우
- **미리** 최대한 분석하고 싶다.

미리보기

$$\llbracket e \rrbracket(\Sigma_0 \infty \Sigma) \subseteq \Sigma_0 \infty \llbracket e \rrbracket \Sigma$$

- Σ_0 : 내가 모르던 외부 환경
- $\llbracket e \rrbracket \Sigma$: 몰라도 최선을 다해 실행한 결과
- ∞ : 합치기(linking) 연산

$$\text{따로분석} = \text{안전한 } \llbracket \cdot \rrbracket^{\#} + \text{안전한 } \infty^{\#}$$

$$\text{분석의 안전성 : } \llbracket e \rrbracket(\Sigma_0 \infty \Sigma) \subseteq \gamma(\overbrace{\sigma_0^{\#} \infty^{\#} \underbrace{\llbracket e \rrbracket^{\#} \sigma^{\#}}_{\text{따로분석}}}^{\text{나중에 합치기}})$$

언어 소개

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

의미공간 소개

$$
\begin{array}{rccll}
\text{Environment} & \sigma & \in & \text{Env} & \\
\text{Location} & \ell & \in & \text{Loc} & \\
\text{Value} & v & \in & \text{Val} \triangleq \text{Env} + \text{Var} \times \text{Expr} \times \text{Env} & \\
\text{Weak Value} & w & \in & \text{WVal} \triangleq \text{Val} + \underline{\text{Val}} & \\
\text{Environment} & \sigma & \rightarrow & \bullet & \text{empty stack} \\
 & & \mid & (x, w) :: \sigma & \text{weak value binding} \\
 & & \mid & (x, \ell) :: \sigma & \text{free location binding} \\
\text{Value} & v & \rightarrow & \sigma & \text{exported environment} \\
 & & \mid & \langle \lambda x.e, \sigma \rangle & \text{closure} \\
\text{Weak Value} & w & \rightarrow & v & \text{value} \\
 & & \mid & \mu\ell.v & \text{recursive value}
\end{array}
$$

의미공간 확장

$$
\begin{array}{rclll}
\text{Event} & E & \rightarrow & \text{Init} & \text{initial environment} \\
 & & | & \text{Read}(E, x) & \text{read event} \\
 & & | & \text{Call}(E, v) & \text{call event} \\
\text{Environment} & \sigma & \rightarrow & \cdots & \\
 & & | & [E] & \text{answer to an event} \\
\text{Value} & v & \rightarrow & \cdots & \\
 & & | & E & \text{answer to an event}
\end{array}
$$

실행의미 소개

$$\boxed{\sigma \vdash e \Downarrow v}$$

$$\textsc{Id}\ \frac{\sigma(x) = v}{\sigma \vdash x \Downarrow v} \qquad \textsc{RecId}\ \frac{\sigma(x) = \mu\ell.v}{\sigma \vdash x \Downarrow v[\mu\ell.v/\ell]} \qquad \textsc{Fn}\ \frac{}{\sigma \vdash \lambda x.e \Downarrow \langle \lambda x.e, \sigma \rangle}$$

$$\textsc{App}\ \frac{\sigma \vdash e_1 \Downarrow \langle \lambda x.e, \sigma_1 \rangle \qquad \sigma \vdash e_2 \Downarrow v_2 \qquad (x, v_2) :: \sigma_1 \vdash e \Downarrow v}{\sigma \vdash e_1\ e_2 \Downarrow v}$$

$$\textsc{Link}\ \frac{\sigma \vdash e_1 \Downarrow \sigma_1 \qquad \sigma_1 \vdash e_2 \Downarrow v}{\sigma \vdash e_1 \rtimes e_2 \Downarrow v} \qquad \textsc{Empty}\ \frac{}{\sigma \vdash \varepsilon \Downarrow \bullet}$$

$$\textsc{Bind}\ \frac{\ell \notin \mathsf{FLoc}(\sigma) \qquad (x, \ell) :: \sigma \vdash e_1 \Downarrow v_1 \qquad (x, \mu\ell.v_1) :: \sigma \vdash e_1 \Downarrow \sigma_2}{\sigma \vdash x = e_1; e_2 \Downarrow (x, \mu\ell.v_1) :: \sigma_2}$$

실행의미 확장

$$\frac{\sigma \vdash e_1 \Downarrow E \qquad [E] \vdash e_2 \Downarrow v}{\sigma \vdash e_1 \rtimes e_2 \Downarrow v}\ \textsc{LinkEvent} \qquad \frac{\sigma \vdash e_1 \Downarrow E \qquad \sigma \vdash e_2 \Downarrow v}{\sigma \vdash e_1\ e_2 \Downarrow \mathsf{Call}(E, v)}\ \textsc{AppEvent}$$

$$\frac{\ell \notin \mathsf{FLoc}(\sigma) \qquad (x, \ell) :: \sigma \vdash e_1 \Downarrow v_1 \qquad (x, \mu\ell.v_1) :: \sigma \vdash e_1 \Downarrow E_2}{\sigma \vdash x = e_1; e_2 \Downarrow (x, \mu\ell.v_1) :: [E_2]}\ \textsc{BindEvent}$$

합치기

$$\boxed{\sigma_0 \infty \cdot \in \mathsf{Event} \to 2^{\mathsf{Val}}}$$

$$\begin{aligned}
\sigma_0 \infty \mathsf{Init} &\triangleq \{\sigma_0\} \\
\sigma_0 \infty \mathsf{Read}(E, x) &\triangleq \{v_+ | \sigma_+ \in \sigma_0 \infty E \wedge \sigma_+(x) = v_+\} \\
&\cup \{v_+[\mu\ell.v_+/\ell] | \sigma_+ \in \sigma_0 \infty E \wedge \sigma_+(x) = \mu\ell.v_+\} \\
\sigma_0 \infty \mathsf{Call}(E, v) &\triangleq \{v_+' | \langle \lambda x.e, \sigma_+\rangle \in \sigma_0 \infty E \wedge v_+ \in \sigma_0 \infty v \wedge (x, v_+) :: \sigma_+ \vdash e \Downarrow v_+'\} \\
&\cup \{\mathsf{Call}(E_+, v_+) | E_+ \in \sigma_0 \infty E \wedge v_+ \in \sigma_0 \infty v\}
\end{aligned}$$

$$\boxed{\sigma_0 \infty \cdot \in \mathsf{Env} \to 2^{\mathsf{Env}}}$$

$$\dots$$

$$\boxed{\sigma_0 \infty \cdot \in \mathsf{Val} \to 2^{\mathsf{Val}}}$$

$$\dots$$

$$\boxed{\sigma_0 \infty \cdot \in \mathsf{WVal} \to 2^{\mathsf{WVal}}}$$

$$\sigma_0 \infty \mu\ell.v \triangleq \{\mu\ell'.v_+ | \ell' \notin \mathsf{FLoc}(v) \cup \mathsf{FLoc}(\sigma_0) \wedge v_+ \in \sigma_0 \infty v[\ell'/\ell]\}$$

Advance

$$\mathrm{eval}(e,\sigma) \triangleq \{v | \sigma \vdash e \Downarrow v\} \quad \mathrm{eval}(e,\Sigma) \triangleq \bigcup_{\sigma\in\Sigma} \mathrm{eval}(e,\sigma) \quad \Sigma_0 \infty W \triangleq \bigcup_{\substack{\sigma_0\in\Sigma_0 \\ w\in W}} (\sigma_0 \infty w)$$

Theorem (Advance)

$$\mathrm{eval}(e, \Sigma_0 \infty \Sigma) \subseteq \Sigma_0 \infty \mathrm{eval}(e,\Sigma)$$

- 증명은 $\sigma \vdash e \Downarrow v$에 대한 귀납법으로
- Coq으로 엄검증 완료! ✓

따로분석

1. 증가하는 $\gamma \in 2^{\mathsf{WVal}} \to \mathsf{WVal}^{\#}$
2. 안전한 $\mathsf{eval}^{\#}$: $\Sigma_0 \subseteq \gamma(\sigma_0^{\#}) \Rightarrow \mathsf{eval}(e, \Sigma_0) \subseteq \gamma(\mathsf{eval}^{\#}(e, \sigma_0^{\#}))$
3. 안전한 $\infty^{\#}$: $\Sigma_0 \subseteq \gamma(\sigma_0^{\#})$ 이고 $W \subseteq \gamma(w^{\#}) \Rightarrow \Sigma_0 \infty W \subseteq \gamma(\sigma_0^{\#} \infty^{\#} w^{\#})$

$\Sigma_0 \subseteq \gamma(\sigma_0^{\#})$ 이고 $\Sigma \subseteq \gamma(\sigma^{\#}) \Rightarrow \mathsf{eval}(e, \Sigma_0 \infty \Sigma) \subseteq \gamma(\sigma_0^{\#} \infty^{\#} \mathsf{eval}^{\#}(e, \sigma^{\#}))$

특히

$$\Sigma_0 \subseteq \gamma(\sigma_0^\#)$$

이고

$$[\mathsf{Init}] \in \gamma(\mathsf{Init}^\#)$$

이면

$$\mathsf{eval}(e_1 \rtimes e_2, \Sigma_0) \subseteq \gamma(\underbrace{\mathsf{eval}^\#(e_1, \sigma_0^\#)}_{\text{따로}} \rtimes^\# \underbrace{\mathsf{eval}^\#(e_2, \mathsf{Init}^\#)}_{\text{따로}})$$

예시: CFA

$$
\begin{array}{rrcl}
\text{Program point} & p & \in & \mathbb{P} \\
\text{Abstract event} & E^{\#} & \in & \mathsf{Event}^{\#} \\
\text{Abstract environment} & \sigma^{\#} & \in & \mathsf{Env}^{\#} \triangleq (\mathsf{Var} \xrightarrow{\mathsf{fin}} 2^{\mathbb{P}}) \times 2^{\mathsf{Event}^{\#}} \\
\text{Abstract closure} & \langle \lambda x.p, p' \rangle & \in & \mathsf{Clos}^{\#} \triangleq \mathsf{Var} \times \mathbb{P} \times \mathbb{P} \\
\text{Abstract value} & v^{\#} & \in & \mathsf{Val}^{\#} \triangleq \mathsf{Env}^{\#} \times 2^{\mathsf{Clos}^{\#}} \\
\text{Abstract semantics} & t^{\#} & \in & \mathbb{T}^{\#} \triangleq \mathbb{P} \to \mathsf{Env}^{\#} \times \mathsf{Val}^{\#} \\
\text{Abstract event} & E^{\#} & \to & \mathsf{Init}^{\#} \mid \mathsf{Read}^{\#}(p, x) \mid \mathsf{Call}^{\#}(p, p)
\end{array}
$$

$$t^\# = p \mapsto (\sigma^\#, v^\#)$$

모든 p마다, 입력 $\sigma^\#$와 출력 $v^\#$가 달려있음.

$$v^\# = (\underbrace{(x \mapsto \{p\}, \overbrace{\{\mathsf{Init}^\#, \mathsf{Read}^\#(p, x), \mathsf{Call}^\#(p_1, p_2)\}}^{E^\# \text{ 부분}})}_{\sigma^\# \text{ 부분}}, \underbrace{\{\langle \lambda x.p, p' \rangle\}}_{\mathsf{Clos}^\# \text{ 부분}})$$

$x \mapsto \{p\}$: $\sigma(x)$가 ℓ^p이거나 p의 출력.

$\mathsf{Read}^\#(p, x)$: p에 들어오는 $[E]$에서 x를 읽음.

$\mathsf{Call}^\#(p_1, p_2)$: p_1에서 나온 E를 p_2에서 나온 v에 적용.

$\langle \lambda x.p, p' \rangle$: p'에 들어온 σ에서 실행되는 함수.

γ 정의하기

$$\boxed{\sigma \preceq (\sigma^\#, t^\#)}$$

$$\frac{}{\bullet \preceq \sigma^\#}\,\textsc{Conc-Nil} \qquad \frac{E \preceq (\sigma^\#, \emptyset)}{[E] \preceq \sigma^\#}\,\textsc{Conc-ENil} \qquad \frac{p \in \sigma^\#.1(x) \qquad \sigma \preceq \sigma^\#}{(x, \ell^p) :: \sigma \preceq \sigma^\#}\,\textsc{Conc-ConsLoc} \qquad \frac{p \in \sigma^\#.1(x) \qquad w \preceq t^\#(p).2 \qquad \sigma \preceq \sigma^\#}{(x, w) :: \sigma \preceq \sigma^\#}\,\textsc{Conc-ConsWVal}$$

$$\boxed{w \preceq (v^\#, t^\#)}$$

$$\frac{\langle \lambda x.p, p' \rangle \in v^\#.2 \qquad \sigma \preceq t^\#(p').1}{\langle \lambda x.p, \sigma \rangle \preceq v^\#}\,\textsc{Conc-Clos} \qquad \frac{v \preceq t^\#(p).2 \qquad v \preceq v^\#}{\mu \ell^p.v \preceq v^\#}\,\textsc{Conc-Rec} \qquad \frac{\mathsf{Init}^\# \in v^\#.1.2}{\mathsf{Init} \preceq v^\#}\,\textsc{Conc-Init}$$

$$\frac{\mathsf{Read}^\#(p, x) \in v^\#.1.2 \qquad [E] \preceq t^\#(p).1}{\mathsf{Read}(E, x) \preceq v^\#}\,\textsc{Conc-Read} \qquad \frac{\mathsf{Call}^\#(p_1, p_2) \in v^\#.1.2 \qquad E \preceq t^\#(p_1).2 \qquad v \preceq t^\#(p_2).2}{\mathsf{Call}(E, v) \preceq v^\#}\,\textsc{Conc-Call}$$

γ 정의하기

$$\gamma(v^\#, t^\#) \triangleq \{w | w \preceq (v^\#, t^\#)\}$$

안전한 Step$^\#$

- $t^\# \sqsupseteq \mathrm{Step}^\#(t^\#)$ 이면

$$\sigma \vdash p \Downarrow v \Rightarrow \sigma \in \gamma(t^\#(p).1, t^\#) \Rightarrow v \in \gamma(t^\#(p).2, t^\#)$$

가 성립하는

안전한 $\text{Link}^{\#}$

- $\sigma_0 \in \gamma(\sigma_0^{\#}, t_0^{\#})$ 인 $\sigma_0^{\#}, t_0^{\#}$ 와
- $t^{\#}$ 가 주어졌을 때,
- $t_+^{\#} \sqsupseteq t_0^{\#}$ 이고 $t_+^{\#} \sqsupseteq \text{Link}^{\#}(\sigma_0^{\#}, t^{\#}, t_+^{\#})$ 이면

$$w_+ \in \sigma_0 \propto w \Rightarrow w \in \gamma(t^{\#}(p).1, t^{\#}) \Rightarrow w_+ \in \gamma(t_+^{\#}(p).1, t_+^{\#})$$

와

$$w_+ \in \sigma_0 \propto w \Rightarrow w \in \gamma(t^{\#}(p).2, t^{\#}) \Rightarrow w_+ \in \gamma(t_+^{\#}(p).2, t_+^{\#})$$

가 성립하는

안전한 $[\![p_0]\!]^\#$ 와 $\leadsto^\#$

$$[\![p_0]\!]^\#(\sigma_0^\#, t_0^\#) \triangleq \mathsf{lfp}(\lambda t^\#.\mathsf{Step}^\#(t^\#) \sqcup [p_0 \mapsto (\sigma_0^\#, \bot)] \sqcup t_0^\#)$$

$$(\sigma_0^\#, t_0^\#) \leadsto^\# t^\# \triangleq \mathsf{lfp}(\lambda t_+^\#.\mathsf{Link}^\#(\sigma_0^\#, t^\#, t_+^\#) \sqcup t_0^\#)$$

감사합니다