

# SNU 4541.664A Program Analysis Spring 2005 Note 21

Prof. Kwangkeun Yi

## 집합 제약식 분석 Constraint-based Analysis

집합 제약식

집합 제약식 분석

집합 제약식 분석의 증명

분석 예

# 집합 제약식 분석의 증명

증명할 부담이 크다:

- 집합 제약식 집합을 도출하는 방법

$$pgm \triangleright C$$

이 옳은가?

- 분석의 해는 규칙  $R$ (새로운 제약식을 더해가는 규칙)을 끝까지 적용해서 나온  $R^*(C)$

$$R^*(C) \stackrel{\text{let}}{=} \text{lfp} \lambda X. C \cup \{a \mid X \vdash_R a\}$$

중에서 완전히 풀려진 제약식들(*atomic constraints*)  
 $\text{explicit}(R^*(C))$ 로 하는데, 옳은가?

# 제약시 도출 방법 $pgm \triangleright C$ 는 옳은가?

$C$ 의 임의의 모델/해(model)가  $pgm$ 의 실제상황을 모두 포섭하는가?

- $\llbracket pgm \rrbracket \in V \rightarrow 2^H$ 를 정의.
  - $H$ 는 분석에서 사용하는 값들의 공간 즉, 구성자(constructor)들로 만들어지는 낱말(term)들의 집합(Herbrand universe)
- $C$ 의 임의의 모델  $\sigma \in V \rightarrow 2^H$ 이  $\llbracket pgm \rrbracket$ 를 포섭함을 증명:

$$\forall x \in V. \llbracket pgm \rrbracket(x) \subseteq \sigma(x)$$

# 제약식 풀기 $explicit(R^*(C))$ 는 옳은가?

- $R^*(C)$ 는 규칙  $R$ 을  $C$ 에 점화해서 변화가 없을 때 까지 적용한다

$$R^*(C) \stackrel{\text{let}}{=} lfp \lambda X. C \cup \{a \mid X \vdash_R a\}$$

끝나는가?

- $\lambda X. C \cup \{a \mid X \vdash_R a\}$  는 제약식 집합을 늘리는 단조함수
- 제약식 집합의 최대 크기는 분석할 프로그램에 의해 한정된다
- 완전히 풀려진 제약식들(*atomic constraints*)  
 $explicit(R^*(C))$ 의 임의의 모델이  $C$ 의 모델인가?

논문읽기

# 분석 예

ML---

$$e \rightarrow x \mid \lambda x.e \mid \mathbf{fix} x.e \mid e e$$

$$\mid \kappa(e) \mid \mathbf{case}(e, \kappa(x) : e, \_ (y) : e)$$

분석 목표: 프로그램에서 각 식들이 계산하는 값들의 집합.  
 집합제약식  $\varphi \supseteq se$

$\varphi \in$	$V = V_e \cup V_x$	프로그램 식과 프로그램 변수마다
$\kappa \in$	$C$	구성자( <i>constructor</i> ) 집합
$se \rightarrow$	$\varphi$	집합변수
	$\lambda x.e$	프로그램 함수
	$\kappa(\varphi)$	구성( <i>construction</i> )
	$\kappa^{-1}(\varphi)$	파괴( <i>deconstruction</i> )
	$se \cap se$	교집합
	$\top \mid \perp$	

제약식 도출  $e \triangleright C$

$$\overline{x \triangleright \{\varphi \supseteq \varphi_x\}}$$

$$\frac{e \triangleright C}{\lambda x. e \triangleright \{\varphi \supseteq \lambda x. e\} \cup C}$$

$$\frac{e \triangleright C}{\text{fix } x. e \triangleright \{\varphi \supseteq \varphi_x, \varphi_x \supseteq \varphi_e\} \cup C}$$

$$\frac{e_1 \triangleright C_1 \quad e_2 \triangleright C_2}{e_1 e_2 \triangleright \left\{ \begin{array}{l} \varphi \supseteq (\varphi_1 \cap \lambda x. e \Rightarrow \varphi_e), \\ \varphi_x \supseteq (\varphi_1 \cap \lambda x. e \Rightarrow \varphi_2) \end{array} \mid \lambda x. e \in \text{pgm} \right\} \cup C_1 \cup C_2}$$

$$\frac{e \triangleright C}{\kappa(e) \triangleright \{\varphi \supseteq \kappa(\varphi_e)\} \cup C}$$

$$\frac{e_0 \triangleright C_0 \quad e_1 \triangleright C_1 \quad e_2 \triangleright C_2}{\text{case}(e_0, \kappa(x) : e_1, -(y) : e_2) \triangleright \begin{array}{l} \{\varphi \supseteq (\varphi_0 \cap \kappa(\top) \Rightarrow \varphi_1), \varphi_x \supseteq (\varphi_0 \cap \kappa(\varphi') \Rightarrow \varphi') \mid \varphi' \in V\} \\ \cup \{\varphi \supseteq (\varphi_0 \cap \kappa'(\top) \Rightarrow \varphi_2), \varphi_y \supseteq (\varphi_0 \cap \kappa'(\varphi') \Rightarrow \varphi') \mid \kappa' \neq \kappa\} \\ \cup C_0 \cup C_1 \cup C_2 \end{array}}$$

제약식 풀기 규칙(*constraint resolving rules*)  $R$ 

$$\frac{\varphi \supseteq \varphi' \quad \varphi' \supseteq se}{\varphi \supseteq se}$$

$$\frac{\varphi \supseteq (\varphi' \cap \lambda x.e \Rightarrow se) \quad \varphi' \supseteq \lambda x.e}{\varphi \supseteq se}$$

$$\frac{\varphi \supseteq (\varphi' \cap \kappa(T) \Rightarrow se) \quad \varphi' \supseteq \kappa(\varphi'')}{\varphi \supseteq se}$$

$$\frac{\varphi \supseteq (\varphi' \cap \kappa(\varphi'') \Rightarrow se) \quad \varphi' \supseteq \kappa(\varphi'')}{\varphi \supseteq se}$$



프로그램  $pgm$ 에서 도출한 제약식 집합  $C$

$$pgm \triangleright C$$

를 끝까지 풀어낸 제약식들  $R^*(C)$

$$R^*(C) \stackrel{\text{let}}{=} lfp \lambda X. C \cup \{a \mid X \vdash_R a\}$$

중에서  $explicit(R^*(C))$ 이  $C$ 의 최소 해/모델.

$$\begin{array}{l} \varphi \supseteq ae \\ ae \rightarrow \top \mid \perp \\ \quad \mid \lambda x.e \\ \quad \mid \kappa(\varphi) \end{array}$$

$$(\mathbf{fix} \ f \ (\lambda x. \mathbf{case}(x, \kappa(k) : k, \_ (y) : f(\kappa(y))))))(\kappa'(\lambda z. z))$$