

# 安全なコード移動が可能な コード生成言語の型システムの設計と実装

大石純平

指導教員 亀山幸義

筑波大学 コンピュータサイエンス専攻

2017/1/27

筑波大学修論審査会

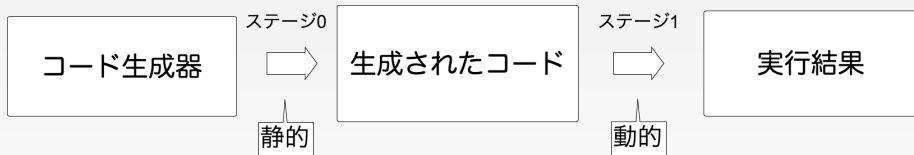
# アウトライン

- ① 目的
- ② 準備
- ③ 問題点
- ④ 解決策
- ⑤ まとめと今後の課題

# アウトライン

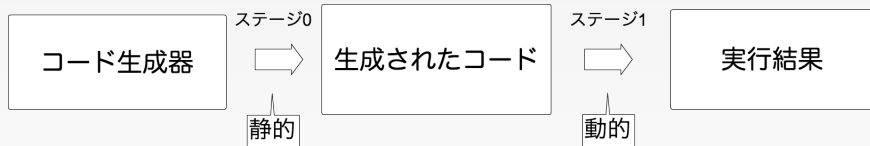
- ① 目的
- ② 準備
- ③ 問題点
- ④ 解決策
- ⑤ まとめと今後の課題

## コード生成の説明と、研究の目的を同時に話す

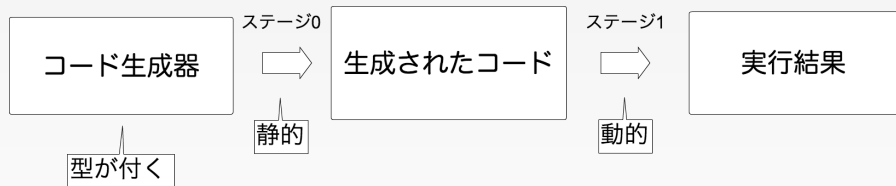


- コード生成をサポートするプログラム言語 (= **コード生成言語**)

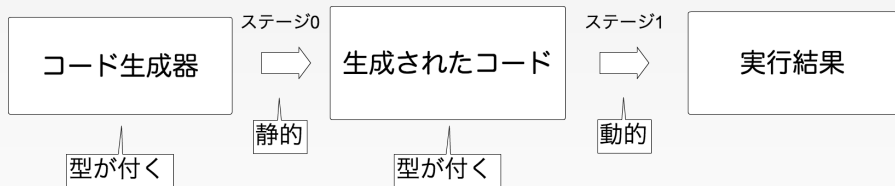
# コード生成前に型付け, 生成後のコードの型安全性を保証



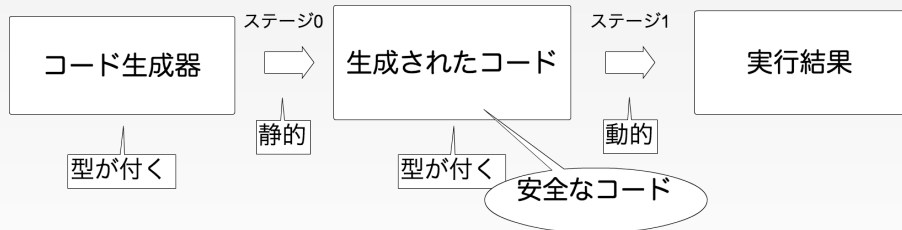
# コード生成前に型付け, 生成後のコードの型安全性を保証



# コード生成前に型付け, 生成後のコードの型安全性を保証

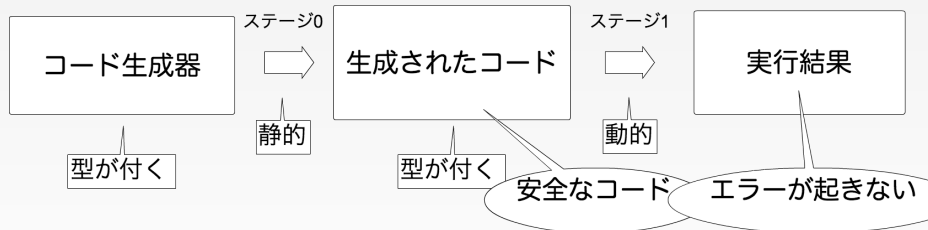


# コード生成前に型付け, 生成後のコードの型安全性を保証





# コード生成前に型付け, 生成後のコードの型安全性を保証



# アウトライン

- ① 目的
- ② 準備
- ③ 問題点
- ④ 解決策
- ⑤ まとめと今後の課題

# コード生成言語による記述例

コード生成器    生成されるコード

$(\underline{\text{int}} \ 3) \rightsquigarrow^* \langle 3 \rangle$

$(\underline{\text{int}} \ 3) \underline{+} (\underline{\text{int}} \ 5) \rightsquigarrow^* \langle 3 + 5 \rangle$

$\underline{\lambda}x. x \underline{+} (\underline{\text{int}} \ 3) \rightsquigarrow^* \langle \lambda x'. x' + 3 \rangle$

$\underline{\text{for}} \ x = \dots \underline{\text{to}} \dots \underline{\text{do}} \dots \rightsquigarrow^* \langle \text{for } x' = \dots \text{ to } \dots \text{ do } \dots \rangle$

# コード生成言語による記述例

コード生成器    生成されるコード

$(\underline{\text{int}} \ 3) \rightsquigarrow^* \langle 3 \rangle$

$(\underline{\text{int}} \ 3) \underline{+} (\underline{\text{int}} \ 5) \rightsquigarrow^* \langle 3 + 5 \rangle$

$\underline{\lambda}x. x \underline{+} (\underline{\text{int}} \ 3) \rightsquigarrow^* \langle \lambda x'. x' + 3 \rangle$

$\underline{\text{for}} \ x = \dots \underline{\text{to}} \dots \underline{\text{do}} \dots \rightsquigarrow^* \langle \text{for } x' = \dots \text{ to } \dots \text{ do } \dots \rangle$

## コードコンビネータ

- 下線付きの演算子
- コードを引数にとり，コードを返す

# let 挿入 (コード移動) の実現方法

## コード生成器

○ for  $x = e1$  to  $e2$  do  
○ for  $y = e3$  to  $e4$  do  
    set  $\langle a \rangle (x, y)$  ○  
    let  $u = cc$  in ○  $u$

$\rightsquigarrow^*$

## 生成されるコード

<let  $u' = cc'$  in  
    for  $x' = e1'$  to  $e2'$  do  
        for  $y' = e3'$  to  $e4'$  do  
             $a[x', y'] \leftarrow u'$ >

# let 挿入 (コード移動) の実現方法

## コード生成器

- for  $x = e1$  to  $e2$  do
- for  $y = e3$  to  $e4$  do  
    set  $\langle a \rangle (x, y)$  ○  
    let  $u = cc$  in ○  $u$

## 生成されるコード

$\rightsquigarrow^*$

< let  $u' = cc'$  in  
    for  $x' = e1'$  to  $e2'$  do  
        for  $y' = e3'$  to  $e4'$  do  
             $a[x', y'] \leftarrow u'$  >

## shift0/reset0 の導入

- のところに **shift0/reset0** 等を用いることで、多段階 let 挿入を行う

# アウトライン

- ① 目的
- ② 準備
- ③ 問題点**
- ④ 解決策
- ⑤ まとめと今後の課題

# コード生成前後でコードが移動する

## コード生成器

for  $x = e1$  to  $e2$  do  
**reset0** for  $y = e3$  to  $e4$  do  
**shift0**  $k \rightarrow$  let  $u = y$  in  
**throw**  $k$  set  $a(x, y) u$

$\rightsquigarrow^*$

## 生成されるコード

< **for**  $x' = e1'$  **to**  $e2'$  **do**  
**let**  $u' = y'$  **in**  
**for**  $y' = e3'$  **to**  $e4'$  **do**  
 $a[(x', y')] \leftarrow u' >$



# コード生成前後でコードが移動する

## コード生成器

for  $x = e1$  to  $e2$  do  
**reset0** for  $y = e3$  to  $e4$  do  $\rightsquigarrow^*$   
**shift0**  $k \rightarrow$  let  $u = y$  in  
**throw**  $k$  set  $a(x, y) u$

## 生成されるコード

< **for**  $x' = e1'$  **to**  $e2'$  **do**  
**let**  $u' = y'$  **in**  
**for**  $y' = e3'$  **to**  $e4'$  **do**  
 $a[(x', y')] \leftarrow u' >$

## Scope Extrusion

(コード移動により) 意図した束縛から、変数が抜け出てしまうこと

# アウトライン

- ① 目的
- ② 準備
- ③ 問題点
- ④ 解決策**
- ⑤ まとめと今後の課題

# 解決策

$\gamma_0$  **for**  $x = e1$  **to**  $e2$  **do**  
 $\gamma_1$  **for**  $y = e3$  **to**  $e4$  **do**  
 $\gamma_2$  **set**  $a(x, y)$   $cc$

スコープ	使えるコード変数
$\gamma_0$	なし
$\gamma_1$	$x$
$\gamma_2$	$x, y$

$$\gamma_2 \geq \gamma_1 \geq \gamma_0$$

# 環境識別子 (EC) を利用したスコープ表現

## 先行研究:

- 局所的なスコープをもつ破壊的変数をもつコード生成の体系に対する (型安全な) 型システムの構築  
[Sudo, Kiselyov, Kameyama 2014]
- グローバルなスコープをもつ破壊的変数への拡張  
[Kiselyov, Kameyama, Sudo 2016]
- コントロールオペレータには非対応

# 環境識別子 (EC) を利用したスコープ表現

## 先行研究:

- 局所的なスコープをもつ破壊的変数をもつコード生成の体系に対する (型安全な) 型システムの構築  
[Sudo, Kiselyov, Kameyama 2014]
- グローバルなスコープをもつ破壊的変数への拡張  
[Kiselyov, Kameyama, Sudo 2016]
- コントロールオペレータには非対応

## 問題点:

shift0/reset0 などのコントロールオペレータは、スコープの包含関係を逆転させてしまう。

# コード生成+shift0/reset0 の型システム (の一部)

reset0:

$$\frac{\Gamma \vdash e : \langle t \rangle^{\gamma} ; \langle t \rangle^{\gamma}, \sigma}{\Gamma \vdash \mathbf{reset0} \ e : \langle t \rangle^{\gamma} ; \sigma}$$

shift0:

$$\frac{\Gamma, k : \langle t1 \rangle^{\gamma1} \Rightarrow \langle t0 \rangle^{\gamma0} \vdash e : \langle t0 \rangle^{\gamma0} ; \sigma \quad \Gamma \models \gamma1 \geq \gamma0}{\Gamma \vdash \mathbf{shift0} \ k \rightarrow e : \langle t1 \rangle^{\gamma1} ; \langle t0 \rangle^{\gamma0}, \sigma}$$

throw:

$$\frac{\Gamma \vdash v : \langle t1 \rangle^{\gamma1 \cup \gamma2} ; \sigma \quad \Gamma \models \gamma2 \geq \gamma0}{\Gamma, k : \langle t1 \rangle^{\gamma1} \Rightarrow \langle t0 \rangle^{\gamma0} \vdash \mathbf{throw} \ k \ v : \langle t0 \rangle^{\gamma2} ; \sigma}$$

# 型付けの例 (1)

$e = \text{reset0} \ (\underline{\text{for}} \ x = e1 \ \underline{\text{to}} \ e2 \ \underline{\text{do}} \\ \text{shift0} \ k \rightarrow \underline{\text{let}} \ u = \boxed{\phantom{x}} \ \underline{\text{in}} \ \text{throw} \ k \ u)$

$$\begin{array}{c}
 \frac{\Gamma b \vdash u : \langle t \rangle^{\gamma_1} \cup \gamma_2; \sigma}{\Gamma b \vdash \text{throw } k \ u : \langle t \rangle^{\gamma_2}; \epsilon} \quad \frac{\Gamma a \vdash \boxed{\phantom{x}} : \langle t \rangle^{\gamma_0}; \epsilon}{\Gamma a \vdash \underline{\text{let}} \ u = \dots : \langle t \rangle^{\gamma_0}; \epsilon} \quad (\gamma_2^*) \\
 \hline
 \frac{\gamma_1 \geq \gamma_0, \ x : \langle t \rangle^{\gamma_1} \vdash \text{shift0 } k \rightarrow \dots : \langle t \rangle^{\gamma_1}; \langle t \rangle^{\gamma_0}}{\vdash \underline{\text{for}} \ x = \dots : \langle t \rangle^{\gamma_0}; \langle t \rangle^{\gamma_0}} \quad (\gamma_1^*) \\
 \hline
 \vdash e : \langle t \rangle^{\gamma_0}; \epsilon
 \end{array}$$

$\Gamma a = \gamma_1 \geq \gamma_0, \ x : \langle t \rangle^{\gamma_1}, \ k : \langle t \rangle^{\gamma_1} \Rightarrow \langle t \rangle^{\gamma_0}$

$\Gamma b = \Gamma a, \ \gamma_2 \geq \gamma_0, \ u : \langle t \rangle^{\gamma_2}$



# 型付けの例 (1)

$e = \text{reset0} \ (\underline{\text{for}} \ x = e1 \ \underline{\text{to}} \ e2 \ \underline{\text{do}}$   
 $\quad \text{shift0 } k \rightarrow \underline{\text{let}} \ u = \boxed{\text{int } 3} \ \underline{\text{in}} \ \text{throw } k \ u)$

$$\begin{array}{c}
 \frac{\Gamma b \vdash u : \langle t \rangle^{\gamma_1} \cup \gamma_2; \sigma}{\Gamma b \vdash \text{throw } k \ u : \langle t \rangle^{\gamma_2}; \epsilon} \quad \vdots \quad \Gamma a \vdash \boxed{\text{int } 3} : \langle t \rangle^{\gamma_0}; \epsilon \\
 \hline
 \Gamma a \vdash \underline{\text{let}} \ u = \dots : \langle t \rangle^{\gamma_0}; \epsilon \quad (\gamma_2^*) \\
 \hline
 \gamma_1 \geq \gamma_0, \ x : \langle t \rangle^{\gamma_1} \vdash \text{shift0 } k \rightarrow \dots : \langle t \rangle^{\gamma_1}; \langle t \rangle^{\gamma_0} \quad (\gamma_1^*) \\
 \hline
 \vdash \underline{\text{for}} \ x = \dots : \langle t \rangle^{\gamma_0}; \langle t \rangle^{\gamma_0} \\
 \hline
 \vdash e : \langle t \rangle^{\gamma_0}; \epsilon
 \end{array}$$

$\Gamma a = \gamma_1 \geq \gamma_0, \ x : \langle t \rangle^{\gamma_1}, \ k : \langle t \rangle^{\gamma_1} \Rightarrow \langle t \rangle^{\gamma_0}$

$\Gamma b = \Gamma a, \ \gamma_2 \geq \gamma_0, \ u : \langle t \rangle^{\gamma_2}$

# 型付けの例 (1)

$e = \text{reset0} \ (\underline{\text{for}} \ x = e1 \ \underline{\text{to}} \ e2 \ \underline{\text{do}} \ \text{shift0} \ k \rightarrow \underline{\text{let}} \ u = \boxed{x \ \underline{+} \ (\underline{\text{int}} \ 3)} \ \underline{\text{in}} \ \text{throw} \ k \ u)$

$$\begin{array}{c}
 \frac{\Gamma b \vdash u : \langle t \rangle^{\gamma_1} \cup \gamma_2; \sigma}{\Gamma b \vdash \text{throw } k \ u : \langle t \rangle^{\gamma_2}; \epsilon} \quad \vdots \quad \Gamma a \vdash \boxed{x \ \underline{+} \ (\underline{\text{int}} \ 3)} : \langle t \rangle^{\gamma_0}; \epsilon \\
 \hline
 \Gamma a \vdash \underline{\text{let}} \ u = \dots : \langle t \rangle^{\gamma_0}; \epsilon \\
 \hline
 \frac{\gamma_1 \geq \gamma_0, \ x : \langle t \rangle^{\gamma_1} \vdash \text{shift0 } k \rightarrow \dots : \langle t \rangle^{\gamma_1}; \langle t \rangle^{\gamma_0}}{\vdash \underline{\text{for}} \ x = \dots : \langle t \rangle^{\gamma_0}; \langle t \rangle^{\gamma_0}} \quad (\gamma_1^*) \\
 \hline
 \vdash e : \langle t \rangle^{\gamma_0}; \epsilon
 \end{array}$$

$\Gamma a = \gamma_1 \geq \gamma_0, \ x : \langle t \rangle^{\gamma_1}, \ k : \langle t \rangle^{\gamma_1} \Rightarrow \langle t \rangle^{\gamma_0}$

$\Gamma b = \Gamma a, \ \gamma_2 \geq \gamma_0, \ u : \langle t \rangle^{\gamma_2}$

# 型付けの例 (2)

$e' = \text{reset0} \text{ (for } x = e1 \text{ to } e2 \text{ do reset0 (for } y = e3 \text{ to } e4 \text{ do}$   
 $\text{shift0 } k_2 \rightarrow \text{shift0 } k_1 \rightarrow \text{let } u = \square \text{ in throw } k_1 \text{ (throw } k_2 \text{ } e5)))$

$$\begin{array}{c}
 \vdots \\
 \hline \Gamma e \vdash e5 : \langle t \rangle^{\gamma 2} \cup \gamma 1 \cup \gamma 3; \quad \epsilon \\
 \hline \Gamma e \vdash \text{throw } k_2 \text{ } e5 : \langle t \rangle^{\gamma 1} \cup \gamma 3; \quad \epsilon \\
 \hline \Gamma e = \Gamma d, \gamma 3 \geq \gamma 0, u : \langle t \rangle^{\gamma 3} \vdash \text{throw } k_1 \dots : \langle t \rangle^{\gamma 3}; \quad \epsilon \quad \Gamma d \vdash \square : \langle t \rangle^{\gamma 0}; \quad \epsilon \quad (\gamma 3^*) \\
 \hline \Gamma d = \Gamma c, k_1 : \langle t \rangle^{\gamma 1} \Rightarrow \langle t \rangle^{\gamma 0} \vdash \text{let } u = \dots : \langle t \rangle^{\gamma 0}; \quad \epsilon \\
 \hline \Gamma c = \Gamma b, k_2 : \langle t \rangle^{\gamma 2} \Rightarrow \langle t \rangle^{\gamma 1} \vdash \text{shift0 } k_1 \dots : \langle t \rangle^{\gamma 1}; \quad \langle t \rangle^{\gamma 0} \\
 \hline \Gamma b = \Gamma a, \gamma 2 \geq \gamma 1, y : \langle t \rangle^{\gamma 2} \vdash \text{shift0 } k_2 \dots : \langle t \rangle^{\gamma 2}; \quad \langle t \rangle^{\gamma 1}, \langle t \rangle^{\gamma 0} \quad (\gamma 2^*) \\
 \hline \Gamma a \vdash \text{for } y = \dots : \langle t \rangle^{\gamma 1}; \quad \langle t \rangle^{\gamma 1}, \langle t \rangle^{\gamma 0} \\
 \hline \Gamma a = \gamma 1 \geq \gamma 0, x : \langle t \rangle^{\gamma 1} \vdash \text{reset0} \dots : \langle t \rangle^{\gamma 1}; \quad \langle t \rangle^{\gamma 0} \quad (\gamma 1^*) \\
 \hline \vdash \text{for } x = \dots : \langle t \rangle^{\gamma 0}; \quad \langle t \rangle^{\gamma 0} \\
 \hline \vdash e' = \text{reset0} \dots : \langle t \rangle^{\gamma 0}; \quad \epsilon
 \end{array}$$

# 型推論アルゴリズム

- 制約生成
- 制約解消

こういう制約が出てくる

# 制約解消

先の制約を解消して，型を決定する

# アウトライン

- ① 目的
- ② 準備
- ③ 問題点
- ④ 解決策
- ⑤ **まとめと今後の課題**

# まとめと今後の課題

## まとめ

- コード生成言語にコード移動を許す仕組み (shift0/reset0) を導入し, その安全性を保証するための型システムの設計を行い
  - 安全性: Scope extrusion が起きないようにする
- 型推論アルゴリズムの開発を行った (実装については制約生成まで)

## 今後の課題

- 設計した型システムの健全性の証明 (Subject reduction) の完成
- 型推論アルゴリズム (制約解消) の実装の完成