依存型を備えた多段階計算の同値型による拡張

勝田峻太朗, 五十嵐淳

京都大学

背景

多段階計算 $\lambda^{\bigcirc}$ [Davies'96]

プログラムを値として表現し，プログラムの生成が可能な計算体系．

- **next**は，コードを生成する．
- **prev**は，コードの中で他のコードを展開する．

$$(\lambda x{:}\bigcirc\mathrm{Int}.\mathbf{next}(3 \times \boxed{\mathbf{prev}(x)}))\ (\mathbf{next}(3+4))$$
$$\to \mathbf{next}(3 \times \boxed{(3+4)})$$

他のコード内部でのコードの展開

研究の動機・概観

動機・目的

多段階計算を含む体系の型システムを強化し，生成したコードの構文木に関する性質を型で保証する．
たとえば，多段階計算におけるコード生成は，計算を速くするために行われることがあり，`pow3_6_code` は，`pow3_6_code_normal` ではなく，より速い `pow3_6_code_fast` の形をしていることを型システムで保証することが考えられる．

```
let gen_pow_code = < something >
(* 3^6のコードを生成 *)
let pow3_6_code = gen_pow_code 3 6
let pow3_6_code_fast =
  next (let x' = 3 * 3 in
    let x'' = x' * x' in
    x'' * x')
let pow3_6_code_normal =
  next(3 * 3 * 3 * 3 * 3 * 3)
```

研究の手法・概観

多段階計算で生成されたコードに関する命題を型で表現できるように，述語論理に対応する依存型の体系 λLF [Harper et al.'93] と，項の同値性を型で表現できる同値型によって体系を拡張した．
※関連研究: λ^{MD} [Kawata&Igarashi'19]

同値型 Eq_T [Martin-Löf'84]

依存型の一種で，2つの項の同値を型として表す．$t_1 \equiv t_2$ であれば，型 $\mathbf{Eq}_T\ t_1\ t_2$ を持つ項が存在する．

$$\frac{\Gamma \vdash^{(n)} T{::}\,* \qquad \Gamma \vdash^{(n)} t{:}T}{\Gamma \vdash^{(n)} \mathbf{id}(t){:}\ \boxed{\mathbf{Eq}_T\ t\ t}}\ (\text{T-EqIntro})$$

同値型の導入

$$\frac{\begin{array}{c}\Gamma \vdash^{(n)} T{::}\,* \qquad \Gamma \vdash^{(n)} a,b{:}T \qquad \Gamma \vdash^{(n)} t_{eq}{:}\ \boxed{\mathbf{Eq}_T\ a\ b} \\ \Gamma, x{:}^{(n)}T, y{:}^{(n)}T, z{:}\mathbf{Eq}_T\ x\ y \vdash^{(n)} C(x,y,z){::}* \\ \Gamma, x{:}^{(n)}T \vdash^{(n)} t{:}C(x,x,id(x))\end{array}}{\Gamma \vdash^{(n)} \mathbf{idpeel}(t_{eq},(x)t){:}C(a,b,t_{eq})}\ (\text{T-EqElim})$$

同値型の使用

同値型に関する型付け規則

研究内容

同値性の定義

コードの内外を捉える

$\boxed{\text{自然数index}}$ を導入し，コード間の同値性はコードの構文木で比べるように定義．

index をインクリメント

$$\frac{\Gamma \vdash^{(n+1)} t_1 \boxed{\equiv_{m+1}} t_2}{\Gamma \vdash^{(n)} \mathbf{next}(t_1) \boxed{\equiv_m} \mathbf{next}(t_2){:}\ \bigcirc T}\ (\text{Q-NextIntro})$$

index をデクリメント

$$\frac{\Gamma \vdash^{(n-1)} t_1 \boxed{\equiv_{m-1}} t_2}{\Gamma \vdash^{(n)} \mathbf{prev}(t_1) \boxed{\equiv_m} \mathbf{prev}(t_2)}\ (\text{Q-PrevIntro})$$

コードでないものしか，β 同値を許さない

$$\frac{\Gamma, x{:}^{(n)}S \vdash^{(n)} t{:}T \qquad \Gamma \vdash^{(n)} s{:}S \qquad \boxed{m=0}}{\Gamma \vdash^{(n)} (\lambda x{:}S.t)s \equiv_{x\mapsto s} t{:}[x \mapsto s]T}\ (\text{Q-Beta})$$

高々一重の **next**の中のみ

$$\frac{\Gamma \vdash^{(n)} t{:}T \qquad \Gamma \vdash^{(n)} T{::}\,* \qquad \boxed{m \le 1}}{\Gamma \vdash^{(n)} \mathbf{prev}(\mathbf{next}(t)) \equiv_m t{:}T}\ (\text{Q-}\bigcirc\text{Beta})$$

$$\frac{\Gamma \vdash^{(n)} t{:}T \qquad \Gamma \vdash^{(n)} T{::}\,* \qquad \boxed{m = 0}}{\Gamma \vdash^{(n)} \mathbf{next}(\mathbf{prev}(t)) \equiv_m t{:}T}\ (\text{Q-}\bigcirc\text{Eta})$$

同値規則の一部

$$3 \equiv_0 (\lambda x{:}\mathrm{Int}.x)\ 3$$
$$\mathbf{next}(3) \not\equiv_0 \mathbf{next}((\lambda x{:}\mathrm{Int}.x)\ 3)$$
$$\mathbf{next}(3 \times \mathbf{prev}(\mathbf{next}(3+4))) \equiv_0 \mathbf{next}(3 \times (3+4))$$
$$\mathbf{next}(\mathbf{next}(3 \times \mathbf{prev}(\mathbf{next}(3+4)))) \not\equiv_0 \mathbf{next}(\mathbf{next}(3 \times (3+4)))$$

同値判断の例

提案した型システムでの表現の例

例1: 型で述語論理の命題を表現する

関数 $f{:}\mathrm{Int} \to \mathrm{Int}$ が冪等であることを型システムで表現することができる．

依存型で表現　同値型で表現

命題	$\forall x$.	$f(x) = f(f(x))$
型	$\Pi x{:}\mathrm{Int}$.	$\mathbf{Eq}_{\mathrm{Int}}\ f(x)\ f(f(x))$

例2: 型でコードの構造を捉える

コード c が，$\mathbf{next}(\lambda x{:}\mathrm{Int}.(3+x))$ の形であることを表現できる．

命題	$c\ =$	$\mathbf{next}(\lambda x{:}\mathrm{Int}.(3+x))$
型	$\mathbf{Eq}_{\bigcirc(\mathrm{Int}\to\mathrm{Int})}\ c$	$\mathbf{next}(\lambda x{:}\mathrm{Int}.(3+x))$

貢献

- 型システム，簡約規則を含む計算体系の設計
- 型安全性の証明
- 合流性の証明
- 強正規化性の証明
- アルゴリズム的型付けと宣言的型付けの同値性の証明

今後の課題

- λ^{MD}にあるような CSP, eval の追加
- Dependent Sum Type の追加