From Parametric Polymorphism to Models of Polymorphic FPC

Rasmus Ejlers Møgelberg

IT University of Copenhagen

PILL$_Y$

- A type theory for domain theory and polymorphism
- DILL + polymorphism + fixed points for terms
- Types:

$$A, B ::= X \mid I \mid A \otimes B \mid A \multimap B \mid !A \mid \prod X\colon \text{Type}.\, A$$

- Intuitionistic function space: $A \to B = !A \multimap B$.
- Terms:

$$\vec{X}\colon \text{Type} \mid \vec{x}\colon \vec{A}; \vec{y}\colon \vec{A}' \vdash M\colon B$$

- Fixed point combinator: $Y\colon \prod X.\, (X \to X) \to X$.

Metalanguage for domain theory

- PILL$_Y$ + parametric polymorphism gives solutions to general recursive type equations
- Recursive type equations satisfy universal condition (initial dialgebra) in *linear* part of calculus.
- Strong reasoning principles for recursive types
- Hypothesis: PILL$_Y$ is a strong metalanguage for domain theory

Testing the hypothesis

- FPC: simply typed lambda calculus + recursive types with call-by-value operational semantics
- Classical domain theory: Sound and computationally adequate model of FPC
- We show how to interpret FPC in models of parametric PILL$_Y$.
- Problem: FPC intuitionistic language, PILL$_Y$ linear
- Extend to polymorphic FPC
- Polymorphic FPC in category of admissible pers.
- Computational adequacy

FPC

- Simply typed lambda calculus + recursive types with call-by-value operational semantics
- Types:

$$\sigma, \tau \quad ::= \quad \alpha \mid 1 \mid \sigma + \tau \mid \sigma \times \tau \mid \sigma \to \tau \mid \mathsf{rec}\ \alpha .\, \sigma$$

- Terms:

$$\begin{aligned} t \quad ::= \quad & x \mid \star \mid \mathsf{inl}\ t \mid \mathsf{inr}\ t \mid \mathsf{case}\ t\ \mathsf{of}\ \mathsf{inl}\ x.\, t'\ \mathsf{of}\ \mathsf{inr}\ x.\, t'' \mid \\ & \langle t, t' \rangle \mid \pi_1(t) \mid \pi_2(t) \mid \lambda x \colon \sigma .\, t \mid t(t') \mid \mathsf{intro}\ t \mid \\ & \mathsf{elim}\ t \mid \mathsf{let\ rec}\ f x = t\ \mathsf{in}\ f\ t' \end{aligned}$$

FPC: Examples of rules

- Recursive types

$$\frac{\vec{\alpha}, \alpha \vdash \tau}{\vec{\alpha} \vdash \mathsf{rec}\ \alpha.\,\tau}$$

$$\frac{\Gamma \vdash_{\vec{\alpha}} t : \mathsf{rec}\ \alpha.\,\tau}{\Gamma \vdash_{\vec{\alpha}} \mathsf{elim}\ (t) : \tau(\mathsf{rec}\ \alpha.\,\tau)}$$

$$\frac{\Gamma \vdash_{\vec{\alpha}} t : \tau(\mathsf{rec}\ \alpha.\,\tau)}{\Gamma \vdash_{\vec{\alpha}} \mathsf{intro}\ (t) : \mathsf{rec}\ \alpha.\,\tau}$$

Operational semantics

- Call-by-value operational semantics
- Values:

$$v ::= \star \mid \mathsf{inl}\ v \mid \mathsf{inr}\ v \mid \langle v, v' \rangle \mid \lambda x\colon \sigma.\, t \mid \mathsf{intro}\ v$$

- Examples of rules

$$\frac{t \Downarrow \lambda x.\, t'' \qquad t' \Downarrow v \qquad t''[v/x] \Downarrow v'}{t(t') \Downarrow v'}$$

$$\frac{t \Downarrow \mathsf{intro}\ v}{\mathsf{elim}\ t \Downarrow v} \qquad \frac{t \Downarrow v}{\mathsf{intro}\ t \Downarrow \mathsf{intro}\ v}$$

Polymorphic FPC

- Adding type: $\prod \alpha.\, \sigma$
- Adding terms: $\Lambda\alpha.\, t, t(\tau)$
- Typing rules:

$$\frac{\vec{x} \colon \vec{\sigma} \vdash_{\vec{\alpha},\alpha} t \colon \tau}{\vec{x} \colon \vec{\sigma} \vdash_{\vec{\alpha}} \Lambda\alpha.\, t \colon \prod \alpha.\, \tau}\ \vec{\alpha} \vdash \vec{\sigma} \qquad \frac{\vec{x} \colon \vec{\sigma} \vdash_{\vec{\alpha}} t \colon \prod \alpha.\, \tau \quad \vec{\alpha} \vdash \tau'}{\vec{x} \colon \vec{\sigma} \vdash_{\vec{\alpha}} t(\tau') \colon \tau[\tau'/\alpha]}$$

- Values: $\Lambda\alpha.\, t$
- Operational semantics

$$\frac{t \Downarrow \Lambda\alpha.\, t' \qquad t'[\tau/\alpha] \Downarrow v}{t(\tau) \Downarrow v}$$

Modelling FPC in domain theory

- Types as predomains
- Terms as partial maps

$$[\![x\colon \sigma \vdash t\colon \tau]\!]\colon [\![\sigma]\!] \to L[\![\tau]\!]$$

- Values as total maps

Modelling FPC in domain theory

- Types as predomains
- Terms as partial maps

$$[\![x\colon \sigma \vdash t\colon \tau]\!]\colon [\![\sigma]\!] \to L[\![\tau]\!]$$

- Values as total maps
- Important adjunction is

$$\mathbf{Cpo} \underset{}{\overset{\top}{\rightleftarrows}} \mathbf{Cpo}_L$$

FPC models from PILL$_\gamma$ models

- PILL$_\gamma$ model $(\mathbf{C}, !)$ axiomatize $\mathbf{Cppo}_\perp$ and lifting comonad.
- Where is $\mathbf{Cpo}$?

FPC models from PILL$_\gamma$ models

- PILL$_\gamma$ model $(\mathbf{C}, !)$ axiomatize $\mathbf{Cppo}_\perp$ and lifting comonad.
- Where is $\mathbf{Cpo}$?
- Consider Eilenberg-Moore category $\mathbf{C}^!$: Objects are $\xi\colon X \to !X$ such that $\epsilon \circ \xi = id_X, !\xi \circ \xi = \delta \circ \xi$.

FPC models from PILL$_\gamma$ models

- PILL$_\gamma$ model $(\mathbf{C}, !)$ axiomatize $\mathbf{Cppo}_\perp$ and lifting comonad.
- Where is $\mathbf{Cpo}$?
- Consider Eilenberg-Moore category $\mathbf{C}^!$: Objects are $\xi : X \to !X$ such that $\epsilon \circ \xi = id_X, !\xi \circ \xi = \delta \circ \xi$.
- Note $(\mathbf{Cppo}_\perp)^! \simeq \mathbf{Cpo}$.

FPC models from PILL$_\gamma$ models

- PILL$_\gamma$ model $(\mathbf{C}, !)$ axiomatize $\mathbf{Cppo}_\perp$ and lifting comonad.
- Where is $\mathbf{Cpo}$?
- Consider Eilenberg-Moore category $\mathbf{C}^!$: Objects are $\xi \colon X \to !X$ such that $\epsilon \circ \xi = id_X, !\xi \circ \xi = \delta \circ \xi$.
- Note $(\mathbf{Cppo}_\perp)^! \simeq \mathbf{Cpo}$.
- **Theorem.** If $(\mathbf{C}, !)$ is a parametric model of PILL$_\gamma$ then FPC can be interpreted soundly in $(\mathbf{C}^!)_L$

$(\mathbf{C}^!)_L$ as FPC model

- **Theorem.** $\mathbf{C}^!$ is cartesian and has Kleisli exponentials (wrt. L).

$(\mathbf{C}^!)_L$ as FPC model

- **Theorem.** $\mathbf{C}^!$ is cartesian and has Kleisli exponentials (wrt. L).
- Recursive types: Need collection of functors $(\mathbf{C}^!)_L{}^{\mathrm{op}} \times (\mathbf{C}^!)_L \to (\mathbf{C}^!)_L$ with fixed points.

$(\mathbf{C}^!)_L$ as FPC model

- **Theorem.** $\mathbf{C}^!$ is cartesian and has Kleisli exponentials (wrt. L).
- Recursive types: Need collection of functors $(\mathbf{C}^!)_L{}^{\mathrm{op}} \times (\mathbf{C}^!)_L \to (\mathbf{C}^!)_L$ with fixed points.
- Have collection of so-called *strong* functors $\mathbf{C}^{\mathrm{op}} \times \mathbf{C} \to \mathbf{C}$ with fixed points.

$(\mathbf{C}^!)_L$ as FPC model

- **Theorem.** $\mathbf{C}^!$ is cartesian and has Kleisli exponentials (wrt. L).
- Recursive types: Need collection of functors $(\mathbf{C}^!)_L{}^{\mathrm{op}} \times (\mathbf{C}^!)_L \to (\mathbf{C}^!)_L$ with fixed points.
- Have collection of so-called *strong* functors $\mathbf{C}^{\mathrm{op}} \times \mathbf{C} \to \mathbf{C}$ with fixed points.
- **Lemma.** $\mathrm{Hom}_{(\mathbf{C}^!)_L}(\xi\colon X \to !X, \xi'\colon Y \to !Y) \cong \mathrm{Hom}_{\mathbf{C}}(X, Y)$.

$(\mathbf{C}^!)_L$ as FPC model

- Model open types as functor pairs

$$\begin{array}{ccc} (\mathbf{C}^!)^{\mathrm{op}} \times \mathbf{C}^! & \xrightarrow{F_{\text{coalg}}} & \mathbf{C}^! \\ \downarrow & & \downarrow \\ \mathbf{C}^{\mathrm{op}} \times \mathbf{C} & \xrightarrow{F} & \mathbf{C} \end{array}$$

$(\mathbf{C}^!)_L$ as FPC model

- Model open types as functor pairs

$$\begin{array}{ccc} (\mathbf{C}^!)^{\mathrm{op}} \times \mathbf{C}^! & \xrightarrow{F_{\text{coalg}}} & \mathbf{C}^! \\ \downarrow & & \downarrow \\ \mathbf{C}^{\mathrm{op}} \times \mathbf{C} & \xrightarrow{F} & \mathbf{C} \end{array}$$

- Such pairs induce functors ${(\mathbf{C}^!)_L}^{\mathrm{op}} \times (\mathbf{C}^!)_L \to (\mathbf{C}^!)_L$.

$(\mathbf{C}^!)_L$ as FPC model

- Model open types as functor pairs

$$\begin{array}{ccc} (\mathbf{C}^!)^{\mathrm{op}} \times \mathbf{C}^! & \xrightarrow{F_{\text{coalg}}} & \mathbf{C}^! \\ \downarrow & & \downarrow \\ \mathbf{C}^{\mathrm{op}} \times \mathbf{C} & \xrightarrow{F} & \mathbf{C} \end{array}$$

- Such pairs induce functors ${(\mathbf{C}^!)_L}^{\mathrm{op}} \times (\mathbf{C}^!)_L \to (\mathbf{C}^!)_L$.
- **Theorem.** If F is strong then Fix$F$ carries a coalgebra structure giving a fixed point for F_{coalg}.

Interpretation of FPC into PILL$_\gamma$

- Semantic analysis defines an interpretation $(–)^\star$ of FPC into PILL$_\gamma$.

Interpretation of FPC into PILL$_Y$

- Semantic analysis defines an interpretation $(-)^\star$ of FPC into PILL$_Y$.
- Interpret FPC types as PILL$_Y$ types plus coalgebra structure.

Interpretation of FPC into PILL$_Y$

- Semantic analysis defines an interpretation $(-)^\star$ of FPC into PILL$_Y$.
- Interpret FPC types as PILL$_Y$ types plus coalgebra structure.
- Extension of Girard's 2nd translation of intuitionistic logic into linear logic, e.g. $(\sigma \to \tau)^\star = !(\sigma^\star \multimap \tau^\star)$

Interpretation of FPC into PILL$_Y$

- Semantic analysis defines an interpretation $(-)^\star$ of FPC into PILL$_Y$.
- Interpret FPC types as PILL$_Y$ types plus coalgebra structure.
- Extension of Girard's 2nd translation of intuitionistic logic into linear logic, e.g. $(\sigma \to \tau)^\star = !(\sigma^\star \multimap \tau^\star)$
- Interpret FPC terms as linear maps in PILL$_Y$.

Interpretation of FPC into PILL$_Y$

- Semantic analysis defines an interpretation $(–)^\star$ of FPC into PILL$_Y$.
- Interpret FPC types as PILL$_Y$ types plus coalgebra structure.
- Extension of Girard's 2nd translation of intuitionistic logic into linear logic, e.g. $(\sigma \to \tau)^\star = !(\sigma^\star \multimap \tau^\star)$
- Interpret FPC terms as linear maps in PILL$_Y$.
- **Theorem (Soundness).** If $t \Downarrow v$ then $t^\star = v^\star$ provable using parametricity.

Interpretation of FPC into PILL$_Y$

- Semantic analysis defines an interpretation $(-)^\star$ of FPC into PILL$_Y$.
- Interpret FPC types as PILL$_Y$ types plus coalgebra structure.
- Extension of Girard's 2nd translation of intuitionistic logic into linear logic, e.g. $(\sigma \to \tau)^\star = !(\sigma^\star \multimap \tau^\star)$
- Interpret FPC terms as linear maps in PILL$_Y$.
- **Theorem (Soundness).** If $t \Downarrow v$ then $t^\star = v^\star$ provable using parametricity.
- **Conjecture.** Interpretation does *not* extend to PolyFPC.

Chain complete pers

- Special case: FPC model based on chain complete pers.
- Suppose D domain, such that $[D \to D]$ retract of D:

$$[D \to D] \underset{\Phi}{\overset{\Psi}{\leftrightarrows}} D$$

  $\Psi \circ \Phi = id_{[D \to D]}$

- Combinatory algebra: $x \cdot y = \Psi(x)(y)$

Chain complete pers II

- Objects: Chain complete partial equivalence relations
- Morphisms: Maps $f\colon D/R \to D/S$ for which there exists a continuous tracker $g\colon D \to D$

$$f([x]_R) = [g(x)]_S$$

- Form category **CCP**.

Domain theory for polymorphism

- **CCP** is cartesian closed

$$R \to S = \{(x, y) \mid \forall z, w.\, zRw \supset x \cdot zSy \cdot w\}$$

- Lifting:

$$L(R) = \{(\bot, \bot)\} \cup \{(\langle \iota, x\rangle, \langle \iota, x'\rangle \mid xRx'\}$$

  where $\iota = \Phi(id_D)$

- Fixed points for all maps $L(R) \to R$
- + products + coproducts + polymorphism
- Many properties in common with category of predomains

Admissible relations

Admissible relation on chain complete pers:

$$A \subseteq D/R \times D/S$$

such that

$$\text{If } \forall n : \mathbb{N}.\, A([x_n],[y_n]) \text{ then } A([\sqcup_n x_n],[\sqcup_n y_n])$$

A model of polymorphic FPC

- Open types interpreted as pairs:
  - $[\![\alpha_1, \ldots, \alpha_n \vdash \sigma]\!]^p \colon \mathbf{CCP}_0^n \to \mathbf{CCP}_0$
  - If $(A_i \colon \text{AdmRel}(R_i, S_i))_{i \leq n}$ then

$$[\![\vec{\alpha} \vdash \sigma]\!]^r(\vec{A}) \colon \text{AdmRel}([\![\vec{\alpha} \vdash \sigma]\!]^p(\vec{R}), [\![\vec{\alpha} \vdash \sigma]\!]^p(\vec{S}))$$

  - Such that

$$[\![\vec{\alpha} \vdash \sigma]\!]^r(eq_{R_1}, \ldots, eq_{R_n}) = eq_{[\![\vec{\alpha} \vdash \sigma]\!]^p(\vec{R})}$$

A parametric model

- Interpretation of function spaces

$$[\![\vec{\alpha} \vdash \sigma \to \tau]\!]^p(\vec{A}) = [\![\vec{\alpha} \vdash \sigma]\!]^p(\vec{A}) \to L([\![\vec{\alpha} \vdash \tau]\!]^p(\vec{A}))$$

- Polymorphism is interpreted parametrically

$$[\![\vec{\alpha} \vdash \textstyle\prod \alpha.\, \sigma]\!]^p(\vec{R}) =$$
$$\{(x, y) \mid \forall S : \mathbf{CCP}_0.\, L[\![\vec{\alpha}, \alpha \vdash \sigma]\!]^p(\vec{R}, S)(x, y) \wedge$$
$$\forall S, S' : \mathbf{CCP}_0.\, \forall A : \mathrm{AdmRel}(S, S').\, L[\![\vec{\alpha}, \alpha \vdash \sigma]\!]^r(e\vec{q}_{\vec{R}}, A)([x], [y])\}$$

Computational adequacy

- **Theorem (Adequacy).** For any PolyFPC program t

$$[\![t]\!] \neq [\bot] \text{ iff } \exists v.\, t \Downarrow v$$

- **Corollary.** If $[\![t]\!] = [\![t']\!]$ then t contextually equivalent to t'.
- **Corollary.** Translation from FPC to PILL$_Y$ is adequate.

Conclusions

- Type theory PILL$_Y$ + parametricity gives models of FPC
- Types of FPC modelled as coalgebras for lifting comonad
- Special case of chain complete pers gives adequate model of full polymorphic FPC
- Computational adequacy allows us to lift reasoning principles from model to logic for Polymorphic FPC
- Problem: Do general PILL$_Y$-models model full polymorphic FPC?