Polymorphic Algebraic Theories

Marcelo Fiore

COMPUTER LABORATORY
UNIVERSITY OF CAMBRIDGE

GlynnFest
1.VI.2013

Joint work with Makoto Hamana

Method and Technique

- Interplay between:

  mathematical structure and formal language.

- Categorical modelling tools:

  categorical algebra, presheaf categories,
  Grothendieck construction, Kan extensions,
  discrete generalised polynomial functors.

This Talk

One more step on ***Algebraic Foundations for Type Theories***.

- Second-order algebraic theories. [CSL'10, MFCS'10]

  (e.g. untyped and simply-typed λ-calculus)

- ▸ Polymorphic algebraic theories. [LICS'13]

  (e.g. System F)

- Dependent algebraic theories.

  (e.g. MLTT)

Polymorphic Algebraic Theories

- Theory:

  Mathematical foundations.

- Application:

  Mechanised formalisation.

Algebraic Desiderata

- Signatures = syntax specification.

Algebraic Desiderata

- Signatures = syntax specification.
- Pre-models = signature algebras.

Algebraic Desiderata

- Signatures = syntax specification.
- Pre-models = signature algebras.
- Free pre-models = abstract syntax.

Algebraic Desiderata

- Signatures = syntax specification.
- Pre-models = signature algebras.
- Free pre-models = abstract syntax.
- Equational presentations = sets of equations.

Algebraic Desiderata

- Signatures = syntax specification.
- Pre-models = signature algebras.
- Free pre-models = abstract syntax.
- Equational presentations = sets of equations.
- Models = pre-models satisfying equational presentations.

Algebraic Desiderata

- Signatures = syntax specification.
- Pre-models = signature algebras.
- Free pre-models = abstract syntax.
- Equational presentations = sets of equations.
- Models = pre-models satisfying equational presentations.
- Logical framework = sound and complete equational logic.

Algebraic Desiderata

- Signatures = syntax specification.
- Pre-models = signature algebras.
- Free pre-models = abstract syntax.
- Equational presentations = sets of equations.
- Models = pre-models satisfying equational presentations.
- Logical framework = sound and complete equational logic.
- Translations = algebraic homomorphisms.

Algebraic Desiderata

- Signatures = syntax specification.
- Pre-models = signature algebras.
- Free pre-models = abstract syntax.
- Equational presentations = sets of equations.
- Models = pre-models satisfying equational presentations.
- Logical framework = sound and complete equational logic.
- Translations = algebraic homomorphisms.
- Equational theories = invariant presentations.

Polymorphic Type Theories

Example: Polymorphic FPC

Types:

$$\tau \quad ::= \quad \alpha \mid \tau_1 + \tau_2 \mid \tau_1 \times \tau_2 \mid \tau_1 \Rightarrow \tau_2 \mid \mu(\alpha.\tau) \mid \forall(\alpha.\tau)$$

Terms:

$$t \quad ::= \quad \cdots \mid \lambda(x : \tau.t) \mid t_1(t_2) \mid \Lambda(\alpha.t) \mid t(\tau)$$

Polymorphic Type Theories

Example: Polymorphic FPC

Types: LEVEL 1

$$\tau \quad ::= \quad \alpha \mid \tau_1 + \tau_2 \mid \tau_1 \times \tau_2 \mid \tau_1 \Rightarrow \tau_2 \mid \mu(\alpha.\tau) \mid \forall(\alpha.\tau)$$

Terms: LEVEL 2

$$t \quad ::= \quad \cdots \mid \lambda(x : \tau.t) \mid t_1(t_2) \mid \Lambda(\alpha.t) \mid t(\tau)$$

Polymorphic Type Theories

Example: Polymorphic FPC with equi-recursive types

Types: LEVEL 1

$$\tau \quad ::= \quad \alpha \mid \tau_1 + \tau_2 \mid \tau_1 \times \tau_2 \mid \tau_1 \Rightarrow \tau_2 \mid \mu(\alpha.\tau) \mid \forall(\alpha.\tau)$$

$$\mu(\alpha.\tau) = \tau[\mu(\alpha.\tau)/\alpha]$$

Terms: LEVEL 2

$$t \quad ::= \quad \cdots \mid \lambda(x : \tau.t) \mid t_1(t_2) \mid \Lambda(\alpha.t) \mid t(\tau)$$

Second-Order Algebraic Theories

- Binding signatures.

Example:

operator		arity	specification
λ	:	$(*)* \longrightarrow *$	one binding argument
@	:	$*, * \longrightarrow *$	two non-binding arguments

Second-Order Algebraic Theories

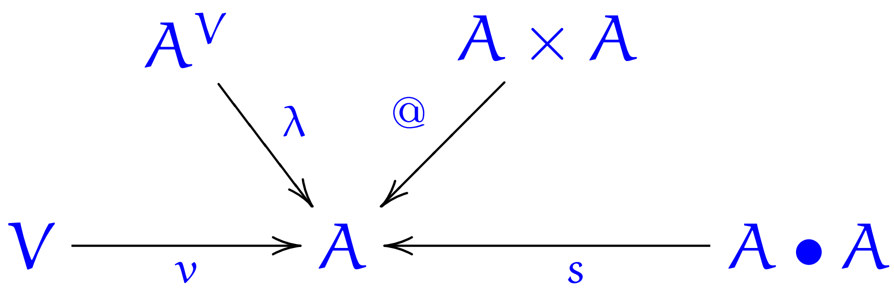
- Binding signatures.

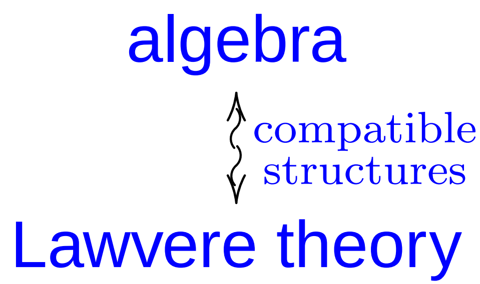
  Example:

operator		arity	specification
λ	:	$(*)* \longrightarrow *$	one binding argument
@	:	$*, * \longrightarrow *$	two non-binding arguments

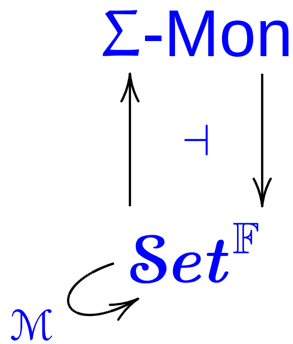
- Pre-models = Σ-monoids in $\mathcal{S}et^{\mathbb{F}}$.

  Example:

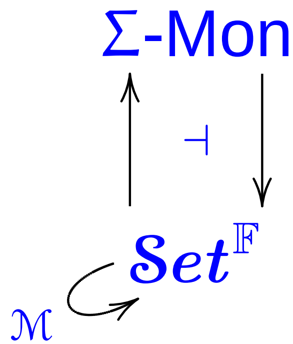




- Free pre-models
  = abstract syntax with variable binding and metavariables



- Free pre-models
  = abstract syntax with variable binding and metavariables



Example: $t \in \mathcal{M}(\underline{X})(\Gamma)$ iff $X \triangleright \Gamma \vdash t$

$$\frac{X \triangleright \Gamma, x \vdash t}{X \triangleright \Gamma \vdash \lambda(x.t)} \qquad \frac{X \triangleright \Gamma \vdash t_1 \quad \Gamma \vdash t_2}{X \triangleright \Gamma \vdash t_1 @ t_2}$$

$$\frac{}{X \triangleright \Gamma \vdash x}\ (x \in \Gamma) \qquad \frac{X \triangleright \Gamma \vdash t_i (1 \leq i \leq n)}{X \triangleright \Gamma \vdash \mathsf{M}[t_1, \cdots, t_n]}\ (\mathsf{M} \in X(n))$$

- Second-order equational presentations

Example:

$$(\beta) \quad \mathsf{M} : [*]* \ , \ \mathsf{N} : * \ \vdash \ \lambda(x.\mathsf{M}[x]) \ @ \ \mathsf{N} \ = \ \mathsf{M}[\mathsf{N}]$$

- Second-order equational presentations

  Example:

$$(\beta)\quad \mathsf{M} : [*]* \;,\; \mathsf{N} : * \;\vdash\; \lambda(x.\mathsf{M}[x])\;@\;\mathsf{N} \;=\; \mathsf{M}[\mathsf{N}]$$

- Second-order models

  Example: premodels A satisfying

$$[\![\, \lambda(x.\mathsf{M}[x])\;@\;\mathsf{N}[\,]\,]\!] \;=\; [\![\, \mathsf{M}[\mathsf{N}[\,]]\,]\!] \;:\; A^V \times A \to A$$

Coincides with Martin Hyland's notion of *semiclosed algebraic theory* as a model of the λ-calculus

- Second-Order Equational Logic

  Judgements

$$X \triangleright \Gamma \vdash t_1 = t_2$$

  subject to congruence rules of meta substitution and extension.

Polymorphic Algebraic Theories

Types: LEVEL 1

Second-order equational presentations

Example:

$$+, \times, \Rightarrow \ : \ *, * \to * \qquad\qquad \forall, \exists, \mu \ : \ (*)* \to *$$

$$\tau : [*]* \ \vdash \ \mu(x.\tau[x]) \ = \ \tau\big[\mu(x.\tau[x])\big]$$

Terms:

LEVEL 2

Example: System F

Vernacular rules.

$$\frac{\alpha_i \mid x_j : \tau_j, x : \sigma \vdash t : \tau}{\alpha_i \mid x_j : \tau_j \vdash \lambda(x : \sigma.t) : \sigma \Rightarrow \tau}$$

$$\frac{\alpha_i, \alpha \mid x_i : \tau_i \vdash t : \tau}{\alpha_i \mid x_i : \tau_i \vdash \Lambda(\alpha.t) : \forall(\alpha.\tau)}$$

Terms: LEVEL 2

Example: System F

Vernacular rules.

$$\frac{\alpha_i \mid x_j : \tau_j, x : \sigma \vdash t : \tau}{\alpha_i \mid x_j : \tau_j \vdash \lambda(x : \sigma.t) : \sigma \Rightarrow \tau}$$

$$\frac{\alpha_i, \alpha \mid x_i : \tau_i \vdash t : \tau}{\alpha_i \mid x_i : \tau_i \vdash \Lambda(\alpha.t) : \forall(\alpha.\tau)}$$

Polymorphic signature operators.

$$\lambda \quad :: \quad \mathsf{S}, \mathsf{T} : * \quad \rhd \quad (\mathsf{S})\mathsf{T} \quad \longrightarrow \quad \mathsf{S} \Rightarrow \mathsf{T}$$

$$\Lambda \quad :: \quad \mathsf{T} : [*]* \quad \rhd \quad \langle\alpha\rangle\mathsf{T}[\alpha] \quad \longrightarrow \quad \forall(\alpha.\mathsf{T}[\alpha])$$

Example: Existential λ-calculus

Vernacular rules.

$$\frac{\alpha_i \mid x_j : \tau_j \vdash s : \sigma[\tau/\alpha]}{\alpha_i \mid x_j : \tau_j \vdash \textsf{pack}(\tau, s) : \exists(\alpha.\sigma)}$$

$$\frac{\alpha_i \mid x_j : \tau_j \vdash s : \exists(\alpha.\sigma) \qquad \alpha_i, \alpha \mid x_j : \tau_j, x : \sigma \vdash t : \tau}{\alpha_i \mid x_j : \tau_j \vdash \textsf{unpack}\ s\ \textsf{as}\ (\alpha, x)\ \textsf{in}\ t : \tau}\ (\alpha \# \tau_j, \tau)$$

Example: Existential λ-calculus

Vernacular rules.

$$\frac{\alpha_i \mid x_j : \tau_j \vdash s : \sigma[\tau/\alpha]}{\alpha_i \mid x_j : \tau_j \vdash \textsf{pack}(\tau, s) : \exists(\alpha.\sigma)}$$

$$\frac{\alpha_i \mid x_j : \tau_j \vdash s : \exists(\alpha.\sigma) \qquad \alpha_i, \alpha \mid x_j : \tau_j, x : \sigma \vdash t : \tau}{\alpha_i \mid x_j : \tau_j \vdash \textsf{unpack}\ s\ \textsf{as}\ (\alpha, x)\ \textsf{in}\ t : \tau} \ (\alpha \# \tau_j, \tau)$$

Polymorphic signature operators.

$$\textsf{pack} \ :: \ \textsf{S} : [*]*, \textsf{T} : * \ \triangleright \ \textsf{S}[\textsf{T}] \longrightarrow \exists(\alpha.\textsf{S}[\alpha])$$

$$\textsf{unpack} \ :: \ \textsf{S} : [*]*, \textsf{T} : * \ \triangleright \ \exists(\alpha.\textsf{S}[\alpha]), \langle\alpha\rangle(\textsf{S}[\alpha])\textsf{T} \longrightarrow \textsf{T}$$

Contexts for Polymorphism

Contexts

$$\alpha_1, \ldots, \alpha_n \;\mid\; \Gamma \;\vdash\; t \;:\; \tau$$

type context | term context | term | type

with types in a universe U are modelled by the Grothendieck construction

[Hamana FOSSACS'11]

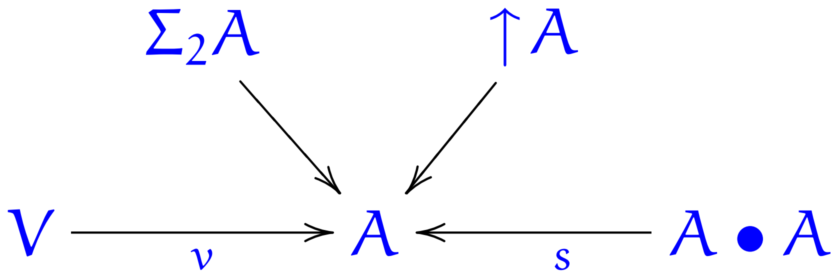
$$\mathsf{GU} = \int^{n \in \mathbb{F}} \mathbb{F}{\downarrow}(\mathsf{U}n) \times \mathsf{U}n$$

Polymorphic Signatures

- A *polymorphic signature* consists of
  1. a second-order signature Σ_1 and equational presentation E_1 for type structure, and
  2. a polymorphic signature Σ_2 for term structure.

Polymorphic Signatures and Structures

- A *polymorphic signature* consists of
  1. a second-order signature Σ_1 and equational presentation E_1 for type structure, and
  2. a polymorphic signature Σ_2 for term structure.
- A *polymorphic structure* consist of
  1. a type universe U modelling (Σ_1, E_1), and
  2. compatible algebraic structure in $\mathcal{S}et^{GU}$ as follows . . .



where Σ_2 and $\uparrow$ are endofunctors on $\mathcal{S}et^{\mathrm{GU}}$ whose algebras respectively model the polymorphic signature operators and the operation of type-in-term substitution.

$$\begin{array}{ccccc} & \Sigma_2 A & & \uparrow A & \\ & & \searrow \quad \swarrow & & \\ V & \xrightarrow[v]{} & A & \xleftarrow[s]{} & A \bullet A \end{array}$$

where Σ_2 and $\uparrow$ are endofunctors on $\mathcal{S}et^{\mathrm{GU}}$ whose algebras respectively model the polymorphic signature operators and the operation of type-in-term substitution.

NB: These constructions are performed within the theory of discrete generalised polynomial functors on presheaf categories[ICALP'12] and hence support free constructions, which provide the abstract syntax of polymorphic algebraic theories.

Polymorphic Equational Presentations

Example: System F

Vernacular.

$$(\beta) \quad \Gamma \ \vdash \ \lambda(x : \sigma.M)@N \ = \ M[N/x] \ : \ \tau$$

$$(\beta') \quad \Gamma \ \vdash \ \Lambda(\alpha.M)\,(\sigma) \ = \ M[\sigma/\alpha] \ : \ \tau[\sigma/\alpha]$$

Formal.

$$(\beta) \quad \mathsf{S}, \mathsf{T} : * \ \rhd \ \mathsf{M} : [\mathsf{S}]\mathsf{T}\,,\ \mathsf{N} : \mathsf{S}$$

$$\vdash \ @_{\mathsf{S},\mathsf{T}}(\,\lambda_{\mathsf{S},\mathsf{T}}(x.\mathsf{M}[x])\,,\,\mathsf{N}\,) \ = \ \mathsf{M}[\mathsf{N}] \ : \ \mathsf{T}$$

$$(\beta') \quad \mathsf{S} : *\,,\ \mathsf{T} : [*]* \ \rhd \ \mathsf{M} : \{\alpha\}\mathsf{T}[\alpha]$$

$$\vdash \ @'_{\mathsf{S},\mathsf{T}}(\,\Lambda_{\mathsf{T}}(\alpha.\mathsf{M}\{\alpha\})\,) \ = \ \mathsf{M}\{\mathsf{S}\} \ : \ \mathsf{T}[\mathsf{S}]$$

Example: Existential λ-calculus.

Vernacular.

$$(\exists\beta)\quad \Gamma \ \vdash \quad \mathsf{unpack}\ \big(\mathsf{pack}(\iota, N)\big)\ \mathsf{as}\ (\alpha, x)\ \mathsf{in}\ M \ = \ M[\iota/\alpha, N/x] \ : \ \tau$$

$$(\exists\eta)\quad \Gamma \ \vdash \ \mathsf{unpack}\ N\ \mathsf{as}\ (\alpha, x)\ \mathsf{in}\ M[\mathsf{pack}(\alpha, x)/z] \ = \ M[N/z] \ : \ \tau$$

Formal.

$$(\exists\beta)\quad \mathsf{S} : [*]* \,,\ \mathsf{T}, \mathsf{U} : * \ \rhd\ \mathsf{M} : \{\alpha\}\big[\mathsf{S}[\alpha]\big]\mathsf{T} \,,\ \mathsf{N} : \mathsf{S}[\mathsf{U}]$$

$$\vdash\ \mathsf{unpack}_{\mathsf{S},\mathsf{T}}(\,\mathsf{pack}_{\mathsf{S},\mathsf{U}}(\mathsf{N})\,,\ \alpha.x.\mathsf{M}\{\alpha\}[x]\,) \ = \ \mathsf{M}\{\mathsf{U}\}[\mathsf{N}] \ : \ \mathsf{T}$$

$$(\exists\eta)\quad \mathsf{S} : [*]* \,,\ \mathsf{T} : * \ \rhd\ \mathsf{M} : \big[\exists(\alpha.\mathsf{S}[\alpha])\big]\mathsf{T} \,,\ \mathsf{N} : \exists(\alpha.\mathsf{S}[\alpha])$$

$$\vdash\ \mathsf{unpack}_{\mathsf{S},\mathsf{T}}(\,\mathsf{N}\,,\ \alpha.x.\mathsf{M}[\mathsf{pack}_{\mathsf{S},\alpha}(x)]\,) \ = \ \mathsf{M}[\mathsf{N}] \ : \ \mathsf{T}$$

Algebraic Models

An *algebraic model* of a polymorphic equational presentation is a polymorphic structure satisfying the axioms.

Example: Seely's PL-category semantics of System F yields an algebraic model of the equational presentation of System F.

Algebraic Models

An *algebraic model* of a polymorphic equational presentation is a polymorphic structure satisfying the axioms.

Example: Seely's PL-category semantics of System F yields an algebraic model of the equational presentation of System F.

Polymorphic Translations

Polymorphic translations $(\mathfrak{U}, A) \to (\mathfrak{U}', A')$ are given by homomorphisms $\varphi : \mathfrak{U} \to \mathfrak{U}'$ and $\vartheta : A \to \varphi^* A'$.

Example: Fujita's CPS translation from System F to the Existential λ-calculus is a polymorphic translation.

Polymorphic Equational Logic

Judgements:

$$Z \;\triangleright\; \vec{\alpha} \;\mid\; \Gamma \;\vdash_{\mathrm{U}}\; s = t \;:\; \tau$$

where U is a type universe, Z is a set of metavariable declarations, $\vec{\alpha} \mid \Gamma \vdash \tau$ is a context, and $Z \triangleright \vec{\alpha} \mid \Gamma \vdash s, t : \tau$ are meta-terms.

Polymorphic Equational Logic

Judgements:

$$Z \;\triangleright\; \vec{\alpha} \;\mid\; \Gamma \;\vdash_{\mathfrak{U}}\; s = t \;:\; \tau$$

where $\mathfrak{U}$ is a type universe, Z is a set of metavariable declarations, $\vec{\alpha} \mid \Gamma \vdash \tau$ is a context, and $Z \triangleright \vec{\alpha} \mid \Gamma \vdash s, t : \tau$ are meta-terms.

Rules:

Congruence of all the algebraic structure plus *universe shift*:

$$\frac{Z \;\triangleright\; \vec{\alpha} \;\mid\; \Gamma \;\vdash_{\mathfrak{U}}\; s = t \;:\; \tau}{(\!|\, Z \,|\!) \;\triangleright\; \vec{\alpha} \;\mid\; (\!|\, \Gamma \,|\!) \;\vdash_{\mathfrak{V}}\; (\!|\, s \,|\!) = (\!|\, t \,|\!) \;:\; (\!|\, \tau \,|\!)} \; (\!|\, - \,|\!) : \mathfrak{U} \to \mathfrak{V}$$