“Parametric Lambda Calculus”

Simona Ronchi Della Rocca

Luca Paolini

ronchi@di.unito.it

Università di Torino

Dipartimento di Informatica

Outline

- Historical Introduction

Outline

- Historical Introduction
- The syntax of the parametric λ calculus

Outline

- Historical Introduction
- The syntax of the parametric λ calculus
- The fundamental properties: Confluence and Standardization

Outline

- Historical Introduction
- The syntax of the parametric λ calculus
- The fundamental properties: Confluence and Standardization
- The Universal Reduction Machine

Outline

- Historical Introduction
- The syntax of the parametric λ calculus
- The fundamental properties: Confluence and Standardization
- The Universal Reduction Machine
- The Operational Semantics

Outline

- Historical Introduction
- The syntax of the parametric λ calculus
- The fundamental properties: Confluence and Standardization
- The Universal Reduction Machine
- The Operational Semantics
- Crossing Call-by-name and Call-by-value

A bit of history

~ 1930 The λ-calculus as logical foundation of mathematics (Curry)

The λ-calculus as intensional language for the computable functions (Curry, Kleene, Turing)

A bit of history

~ 1930 The λ-calculus as logical foundation of mathematics (Curry)
The λ-calculus as intensional language for the computable functions (Curry, Kleene, Turing)

~ 1960 LISP: the first programming language inspired by λ-calculus (MC Carthy)
The translation of Algol 60 into λ-calculus (Landin)
The language CUCH (Böhm and Gross)
The Separability Theorem (Böhm)

A bit of history

~ 1930 The λ-calculus as logical foundation of mathematics (Curry)
The λ-calculus as intensional language for the computable functions (Curry, Kleene, Turing)

~ 1960 LISP: the first programming language inspired by λ-calculus (MC Carthy)
The translation of Algol 60 into λ-calculus (Landin)
The language CUCH (Böhm and Gross)
The Separability Theorem (Böhm)

~ 1970 The denotational semantics of λ-calculus (Scott)
The λ-β_v-calculus (Plotkin)

A bit of history

$80 - 90$ The incompleteness of λ-calculus (Ronchi, Honsell)
The full-abstraction problem (Plotkin, Berry)
The lazy λ-calculus (Abramsky)
Other λ-calculi....

Some λ-calculi

- The classical λ-calculus:
  a paradigmatic language for the call-by-name computations.

Some λ-calculi

- The classical λ-calculus:
  a paradigmatic language for the call-by-name computations.
- The λ-β_v-calculus:
  a paradigmatic language for the lazy-call-by-value computations.

“Historia magistra vitae”

- question:
  Is there a unique paradigmatic language, that can model all the computations listed before (and may be that can suggest new computations)?

“Historia magistra vitae”

- question:
  Is there a unique paradigmatic language, that can model all the computations listed before (and may be that can suggest new computations)?
- answer:
  The parametric λ-calculus.

The parametric λ-calculus

The parametric λ-calculus is an abstract calculus with some basic requirements, parametric with respect to a set of terms. Different instantiations of the parameter produce a different λ-calculus.

The parametric λ-calculus

The parametric λ-calculus is an abstract calculus with some basic requirements, parametric with respect to a set of terms. Different instantiations of the parameter produce a different λ-calculus.

- It allows to generalize definitions, properties and proofs

The parametric λ-calculus

The parametric λ-calculus is an abstract calculus with some basic requirements, parametric with respect to a set of terms. Different instantiations of the parameter produce a different λ-calculus.

- It allows to generalize definitions, properties and proofs
- It is a natural starting point for the study of new calculi

The parametric λ-calculus

The parametric λ-calculus is an abstract calculus with some basic requirements, parametric with respect to a set of terms. Different instantiations of the parameter produce a different λ-calculus.

- It allows to generalize definitions, properties and proofs
- It is a natural starting point for the study of new calculi
- It helps the investigations of the relations between different calculi

The Syntax

- The set Λ is produced by

$$M ::= x \mid MM \mid \lambda x.M$$

The Syntax

- The set Λ is produced by

$$M ::= x \mid MM \mid \lambda x.M$$

- Let $\Delta \subseteq \Lambda$. The Δ-reduction ($\rightarrow_\Delta$) is the contextual closure of the following rule:

$$(\lambda x.M)N \rightarrow_\Delta M[N/x] \text{ if and only if } N \in \Delta$$

Set of Input Values

Let $\Delta \subseteq \Lambda$.

- Δ is a set of input values, when:

Set of Input Values

Let $\Delta \subseteq \Lambda$.

- Δ is a set of input values, when:
  - $\mathrm{Var} \subseteq \Delta$ (variable closure)

Set of Input Values

Let $\Delta \subseteq \Lambda$.

- Δ is a set of input values, when:
  - $\mathrm{Var} \subseteq \Delta$ (variable closure)
  - $P, Q \in \Delta$ implies $P[Q/x] \in \Delta$, (substitution closure)

Set of Input Values

Let $\Delta \subseteq \Lambda$.

- Δ is a set of input values, when:
  - $\mathrm{Var} \subseteq \Delta$ (variable closure)
  - $P, Q \in \Delta$ implies $P[Q/x] \in \Delta$, (substitution closure)
  - $M \in \Delta$ and $M \rightarrow_{\Delta} N$ imply $N \in \Delta$ (reduction closure)

Set of Input Values

Let $\Delta \subseteq \Lambda$.

- Δ is a set of input values, when:
  - $\mathrm{Var} \subseteq \Delta$ (variable closure)
  - $P, Q \in \Delta$ implies $P[Q/x] \in \Delta$, (substitution closure)
  - $M \in \Delta$ and $M \rightarrow_{\Delta} N$ imply $N \in \Delta$ (reduction closure)
- The above requirements assure that the status of being an input value is preserved during the evalution process.

Some $\lambda\Delta$-calculi

- Λ is a set of input values
  (the $\lambda\Lambda$-calculus is the classical λ-calculus)

Some $\lambda\Delta$-calculi

- Λ is a set of input values
  (the $\lambda\Lambda$-calculus is the classical λ-calculus)
- $\Gamma = Var \cup \{\lambda x.M \mid M \in \Lambda\}$ is a set of input values
  (the $\lambda\Gamma$-calculus is the Plotkin's $\lambda\beta_v$-calculus)

Some $\lambda\Delta$-calculi

- Λ is a set of input values
  (the $\lambda\Lambda$-calculus is the classical λ-calculus)
- $\Gamma = Var \cup \{\lambda x.M \mid M \in \Lambda\}$ is a set of input values
  (the $\lambda\Gamma$-calculus is the Plotkin's $\lambda\beta_v$-calculus)
- $\Xi = Var \cup \{M \mid M$ is a closed β normal form $\}$ is a set of input value

Some $\lambda\Delta$-calculi

- Λ is a set of input values
  (the $\lambda\Lambda$-calculus is the classical λ-calculus)
- $\Gamma = Var \cup \{\lambda x.M \mid M \in \Lambda\}$ is a set of input values
  (the $\lambda\Gamma$-calculus is the Plotkin's $\lambda\beta_v$-calculus)
- $\Xi = Var \cup \{M \mid M$ is a closed β normal form $\}$ is a set of input value
- Λ_I, defined by the following grammar:

$$M ::= x \mid MM \mid \lambda x.M (x \in FV(M))$$

  is a set of input values

Non Input Values

- Λ-normal forms are not input values

Non Input Values

- Λ-normal forms are not input values
- Λ-head normal forms are not input values

Non Input Values

- Λ-normal forms are not input values
- Λ-head normal forms are not input values
- Γ-normal forms are not input values

Confluence

$M \to^*_{\Delta} N_1$ and $M \to^*_{\Delta} N_2$ imply there is N_3 such that both $N_1 \to^*_{\Delta} N_3$ and $N_2 \to^*_{\Delta} N_3$.

Confluence

$M \to^*_\Delta N_1$ and $M \to^*_\Delta N_2$ imply there is N_3 such that both $N_1 \to^*_\Delta N_3$ and $N_2 \to^*_\Delta N_3$.

- The closure conditions on the set Δ of input values assure that the corresponding λ-Δ-calculus enjoys the confluence property.
- The closure conditions are sufficient but not necessary for garantee the confluence property (instead of the reduction closure it would be sufficent to have: $M \in \Delta$ and $M \to^*_\Delta N$ imply there is $P \in \Delta$ such that $N \to^*_\Delta P$.)

Confluence

$M \rightarrow^*_\Delta N_1$ and $M \rightarrow^*_\Delta N_2$ imply there is N_3 such that both $N_1 \rightarrow^*_\Delta N_3$ and $N_2 \rightarrow^*_\Delta N_3$.

- The closure conditions on the set Δ of input values assure that the corresponding λ-Δ-calculus enjoys the confluence property.
- The closure conditions are sufficient but not necessary for garantee the confluence property (instead of the reduction closure it would be sufficent to have: $M \in \Delta$ and $M \rightarrow^*_\Delta N$ imply there is $P \in \Delta$ such that $N \rightarrow^*_\Delta P$.)

A counterexample

Let us take Λ-normal forms as input values.

Then the confluence fails, in fact:

$(\lambda x.(\lambda y.z)(x(\lambda x.xx)))(\lambda x.xx)$ reduces to both z and $(\lambda y.z)((\lambda x.xx)(\lambda x.xx))$, which do not have a common reduct.

Standardization

$M \to^*_\Delta N$ implies there is a standard reduction sequence such that $M \to^*_\Delta N$ ($M \to^s_\Delta N$) .

Standardization

$M \to^*_\Delta N$ implies there is a standard reduction sequence such that $M \to^*_\Delta N$ ($M \to^s_\Delta N$) .

In case of $\lambda\Lambda$-calculus a standard reduction sequence is a strategy choosing redexes from left to right. It is not easy to formalize such a strategy for other sets of input values.

Standardization

$M \to^*_\Delta N$ implies there is a standard reduction sequence such that $M \to^*_\Delta N$ ($M \to^s_\Delta N$) .

In case of $\lambda\Lambda$-calculus a standard reduction sequence is a strategy choosing redexes from left to right. It is not easy to formalize such a strategy for other sets of input values.

$$(\lambda x.x(KI))(II) \to_\Lambda II(KI) \to_\Lambda I(KI) \to_\Lambda I(\lambda y.I)$$

is a standard reduction sequence in the $\lambda\Lambda$-calculus.

Standardization

$M \to_{\Delta}^{*} N$ implies there is a standard reduction sequence such that $M \to_{\Delta}^{*} N$ ($M \to_{\Delta}^{s} N$) .

In case of $\lambda\Lambda$-calculus a standard reduction sequence is a strategy choosing redexes from left to right. It is not easy to formalize such a strategy for other sets of input values.

$$(\lambda x.x(KI))(II) \to_{\Lambda} II(KI) \to_{\Lambda} I(KI) \to_{\Lambda} I(\lambda y.I)$$

is a standard reduction sequence in the $\lambda\Lambda$-calculus.

The reduction sequence reducing the same term from left to right in the $\lambda\Gamma$-calculus:

$$(\lambda x.x(KI))(II) \to_{\Gamma} (\lambda x.x(\lambda y.I))(II) \to_{\Gamma} (\lambda x.x(\lambda y.I))I \to_{\Gamma} I(\lambda y.I)$$

is not a standard reduction sequence!

Sequentialization

- The Δ-sequentialization $(M)^\circ$ of a term M is a function from Λ to Λ defined as follows:
  - $(xM_1...M_m)^\circ = x(M_1)^\circ...(M_m)^\circ$;
  - $((\lambda x.P)QM_1...M_m)^\circ = (\lambda x.P)^\circ(Q)^\circ(M_1)^\circ...(M_m)^\circ$ if $Q \in \Delta$;
  - $((\lambda x.P)QM_1...M_m)^\circ = (Q)^\circ(\lambda x.P)^\circ(M_1)^\circ...(M_m)^\circ$ if $Q \notin \Delta$;
  - $(\lambda x.P)^\circ = \lambda x.(P)^\circ$.

Sequentialization

- The Δ-sequentialization $(M)^\circ$ of a term M is a function from Λ to Λ defined as follows:
  - $(xM_1...M_m)^\circ = x(M_1)^\circ...(M_m)^\circ$;
  - $((\lambda x.P)QM_1...M_m)^\circ = (\lambda x.P)^\circ(Q)^\circ(M_1)^\circ...(M_m)^\circ$ if $Q \in \Delta$;
  - $((\lambda x.P)QM_1...M_m)^\circ = (Q)^\circ(\lambda x.P)^\circ(M_1)^\circ...(M_m)^\circ$ if $Q \notin \Delta$;
  - $(\lambda x.P)^\circ = \lambda x.(P)^\circ$.
- The degree of a redex R in M is the numbers of λ's which both are active in M and occur on the left of $(R)^\circ$ in $(M)^\circ$.

  (A symbol λ in a term M is active if and only if it is the first symbol of a Δ-redex of M).

Sequentialization

- A sequence $M \equiv P_0 \rightarrow_\Delta P_1 \rightarrow_\Delta ... \rightarrow_\Delta P_n \rightarrow_\Delta N$ is standard if and only if the degree of the redex contracted in P_i is less than or equal to the degree of the redex contracted in P_{i+1}, for every $i < n$.

Sequentialization

- A sequence $M \equiv P_0 \rightarrow_\Delta P_1 \rightarrow_\Delta \ldots \rightarrow_\Delta P_n \rightarrow_\Delta N$ is standard if and only if the degree of the redex contracted in P_i is less than or equal to the degree of the redex contracted in P_{i+1}, for every $i < n$.
- A set Δ of input values is standard if and only if $M \notin \Delta$ and $M \rightarrow^*_\Delta N$ by reducing at every step a not principal redex imply $N \notin \Delta$.
  (The principal redex is the redex of degree 0
  corresponding to the head redex in the $\lambda\Lambda$-calculus)

Sequentialization

- A sequence $M \equiv P_0 \rightarrow_\Delta P_1 \rightarrow_\Delta \ldots \rightarrow_\Delta P_n \rightarrow_\Delta N$ is standard if and only if the degree of the redex contracted in P_i is less than or equal to the degree of the redex contracted in P_{i+1}, for every $i < n$.
- A set Δ of input values is standard if and only if $M \notin \Delta$ and $M \rightarrow^*_\Delta N$ by reducing at every step a not principal redex imply $N \notin \Delta$.
  (The principal redex is the redex of degree 0 corresponding to the head redex in the $\lambda\Lambda$-calculus)
- A $\lambda\Delta$-calculus is standard if and only if the set Δ is standard

Sequentialization

- A sequence $M \equiv P_0 \to_\Delta P_1 \to_\Delta \ldots \to_\Delta P_n \to_\Delta N$ is standard if and only if the degree of the redex contracted in P_i is less than or equal to the degree of the redex contracted in P_{i+1}, for every $i < n$.
- A set Δ of input values is standard if and only if $M \not\in \Delta$ and $M \to^*_\Delta N$ by reducing at every step a not principal redex imply $N \not\in \Delta$.
  (The principal redex is the redex of degree 0 corresponding to the head redex in the $\lambda\Lambda$-calculus)
- A $\lambda\Delta$-calculus is standard if and only if the set Δ is standard
- To be standard is a necessary and sufficent condition for enjoying the standardization property.

Standardization (examples)

- $(\lambda x.x(KI))(II) \rightarrow_\Gamma (\lambda x.x(KI))I \rightarrow_\Gamma I(KI) \rightarrow_\Gamma I(\lambda y.I)$

  is a standard reduction sequence in the $\lambda\Gamma$-calculus.

Standardization (examples)

- $(\lambda x.x(KI))(II) \rightarrow_\Gamma (\lambda x.x(KI))I \rightarrow_\Gamma I(KI) \rightarrow_\Gamma I(\lambda y.I)$

  is a standard reduction sequence in the $\lambda\Gamma$-calculus.
- The set of input values Λ, Γ, Ξ are standard.

Standardization (examples)

- $(\lambda x.x(KI))(II) \to_\Gamma (\lambda x.x(KI))I \to_\Gamma I(KI) \to_\Gamma I(\lambda y.I)$ is a standard reduction sequence in the $\lambda\Gamma$-calculus.
- The set of input values Λ, Γ, Ξ are standard.
- The set of input values Λ_I is not standard.

Evaluation

- The principal evaluation is a sequence of reduction performing at every step the principal redex.

Evaluation

- The principal evaluation is a sequence of reduction performing at every step the principal redex.
- The principal evaluation is normalizing.

Evaluation

- The principal evaluation is a sequence of reduction performing at every step the principal redex.
- The principal evaluation is normalizing.
- There is a Parametric Principal Reduction Machine, parametric with respect to the set of input values Δ, reducing a term according to the principal evaluation.

The PPR machine

$$\frac{M \rightarrow^p_\Delta N}{\lambda x.M \rightarrow^p_\Delta \lambda x.N}\ p1 \qquad \frac{i = \min\{j \leq m | M_i \notin \Delta\text{-nf}\} \quad M_i \rightarrow^p_\Delta N_i}{xM_1 \ldots M_m \rightarrow^p_\Delta xM_1 \ldots N_i \ldots M_m}\ p2$$

$$\frac{Q \in \Delta}{(\lambda x.P)QM_1 \ldots M_m \rightarrow^p_\Delta P[Q/x]M_1 \ldots M_m}\ p3$$

$$\frac{Q \notin \Delta \quad Q \notin \Delta\text{-nf} \quad Q \rightarrow^p_\Delta Q'}{(\lambda x.P)QM_1 \ldots M_m \rightarrow^p_\Delta (\lambda x.P)Q'M_1 \ldots M_m}\ p4$$

$$\frac{Q \notin \Delta \quad Q \in \Delta\text{-nf} \quad P \notin \Delta\text{-nf} \quad P \rightarrow^p_\Delta P'}{(\lambda x.P)QM_1 \ldots M_m \rightarrow^p_\Delta (\lambda x.P')QM_1 \ldots M_m}\ p5$$

$$\frac{Q \notin \Delta \quad P, Q \in \Delta\text{-nf} \quad i = \min\{j \leq m | M_i \notin \Delta\text{-nf}\} \quad M_i \rightarrow^p_\Delta N_i}{(\lambda x.P)QM_1 \ldots M_m \rightarrow^p_\Delta (\lambda x.P)QM_1 \ldots N_i \ldots M_m}\ p6$$

Output values

- Let Δ be a set of input values.
  A set of output values with respect to Δ is any set $\Theta \subseteq \Lambda$ such that:
  - Θ contains all the Δ-normal forms;
  - If $M =_{\Delta} N$ and $N \in \Theta$ then there is $P \in \Theta$ such that $M \to_{\Delta}^{*p} P$

    ($\to^{*p}$ denotes the reduction sequence choosing at every step the principal redex)

Examples of Output Values

- $\triangle$-normal forms is a set of output values w.r.t $\triangle$, for all $\triangle$.

Examples of Output Values

- Δ-normal forms is a set of output values w.r.t Δ, for all Δ.
- Λ-head normal forms is a set of output values w.r.t Λ.

Examples of Output Values

- Δ-normal forms is a set of output values w.r.t Δ, for all Δ.
- Λ-head normal forms is a set of output values w.r.t Λ.
- Λ-weak head normal forms is a set of output values w.r.t Λ.

Examples of Output Values

- Δ-normal forms is a set of output values w.r.t Δ, for all Δ.
- Λ-head normal forms is a set of output values w.r.t Λ.
- Λ-weak head normal forms is a set of output values w.r.t Λ.
- Γ together with Γ-normal forms are a set of output values w.r.t Γ.

The Operational Semantics

- Let Θ be a set of output values with respect to the set of input value Δ. $\Downarrow_{\Delta,\Theta}$ is the evaluation relation defined through the following rules:

$$\frac{M \in \Theta}{M \Downarrow_{\Delta,\Theta} M}\ (axiom) \qquad\qquad \frac{M \to^p_\Delta P \qquad P \Downarrow_{\Delta,\Theta} N}{M \Downarrow_{\Delta,\Theta} N}\ (eval)$$

The Operational Semantics

- Let Θ be a set of output values with respect to the set of input value Δ. $\Downarrow_{\Delta,\Theta}$ is the evaluation relation defined through the following rules:

$$\frac{M \in \Theta}{M \Downarrow_{\Delta,\Theta} M}\ (axiom) \qquad \frac{M \rightarrow^{p}_{\Delta} P \qquad P \Downarrow_{\Delta,\Theta} N}{M \Downarrow_{\Delta,\Theta} N}\ (eval)$$

- Operational pre-order:
$M \preceq_{\Delta,\Theta} N$ if and only if, for all contexts $C[.]$ such that $C[M], C[N] \in \Lambda^0$, $\big(C[M] \Downarrow_{\Delta,\Theta}$ implies $C[N] \Downarrow_{\Delta,\Theta}\big)$.

The Operational Semantics

- Let Θ be a set of output values with respect to the set of input value Δ. $\Downarrow_{\Delta,\Theta}$ is the evaluation relation defined through the following rules:

$$\frac{M \in \Theta}{M \Downarrow_{\Delta,\Theta} M}\,(axiom) \qquad\qquad \frac{M \rightarrow^{p}_{\Delta} P \qquad P \Downarrow_{\Delta,\Theta} N}{M \Downarrow_{\Delta,\Theta} N}\,(eval)$$

- Operational pre-order:
  $M \preceq_{\Delta,\Theta} N$ if and only if, for all contexts $C[.]$ such that $C[M], C[N] \in \Lambda^0$, $\big(C[M] \Downarrow_{\Delta,\Theta}$ implies $C[N] \Downarrow_{\Delta,\Theta}\big)$.
- Operational Equivalence:
  $M \approx_{\Delta,\Theta} N$ if and only if $M \preceq_{\Delta,\Theta} N$ and $N \preceq_{\Delta,\Theta} M$.

Normal forms: an apparent paradox

- In the $\lambda\Lambda$-calculus, the set of normal forms:

$$M ::= x \mid \lambda\vec{x}.xM...M$$

 is semantically meaningful. The Böhm separability property holds, so two $\beta\eta$-different normal forms cannot be equated.

Normal forms: an apparent paradox

- In the $\lambda\Lambda$-calculus, the set of normal forms:

$$M ::= x \mid \lambda\vec{x}.xM...M$$

 is semantically meaningful. The Böhm separability property holds, so two $\beta\eta$-different normal forms cannot be equated.

- In a generic $\lambda\Delta$-calculus, the set of normal forms:

$$M ::= x \mid \lambda\vec{x}.xM...M \mid \lambda\vec{x}.(\lambda x.P)QM..M$$

$$(Q \notin \Delta)$$

 has not good semantical properties.

Normal forms in $\lambda\Gamma$

- In the $\lambda\Gamma$-calculus, two Γ-normal forms can be consistently equated, for example it is consistent to put:

$$\lambda x.I(xx) = \lambda x.xx.$$

Normal forms in $\lambda\Gamma$

- In the $\lambda\Gamma$-calculus, two Γ-normal forms can be consistently equated, for example it is consistent to put:

$$\lambda x.I(xx) = \lambda x.xx.$$

- Let $D \equiv \lambda x.xx$. $(\lambda y.D)(xI)D$ is a Γ-nf.
  It is consistent to put:

$$(\lambda y.D)(xI)D = DD \qquad (DD \text{ is not a value!})$$

Normal forms in $\lambda\Gamma$

- In the $\lambda\Gamma$-calculus, two Γ-normal forms can be consistently equated, for example it is consistent to put:

$$\lambda x.I(xx) = \lambda x.xx.$$

- Let $D \equiv \lambda x.xx$. $(\lambda y.D)(xI)D$ is a Γ-nf.
It is consistent to put:

$$(\lambda y.D)(xI)D = DD \qquad (DD \text{ is not a value!})$$

- A Separability Theorem holds, saying that two different Λ-normal forms can be Γ-separated, so they are different in every Γ model.

Normal forms in $\lambda\Gamma$

- In the $\lambda\Gamma$-calculus, two Γ-normal forms can be consistently equated, for example it is consistent to put:

$$\lambda x.I(xx) = \lambda x.xx.$$

- Let $D \equiv \lambda x.xx$. $(\lambda y.D)(xI)D$ is a Γ-nf.
It is consistent to put:

$$(\lambda y.D)(xI)D = DD \qquad (DD \text{ is not a value!})$$

- A Separability Theorem holds, saying that two different Λ-normal forms can be Γ-separated, so they are different in every Γ model.
- We conjecture that the same holds for all meaningful Δ.

Solvability & Valuability

- A term M is valuable iff $M \to^*_\Delta N \in \Delta$.

Solvability & Valuability

- A term M is valuable iff $M \to^*_\Delta N \in \Delta$.
- A term M is potentially Δ-valuable iff there is a substitution $\mathbf{s}$ replacing variables by closed values, s.t. $\mathbf{s}(M)$ is Δ-valuable.

Solvability & Valuability

- A term M is valuable iff $M \to_{\Delta}^{*} N \in \Delta$.
- A term M is potentially Δ-valuable iff there is a substitution $\mathbf{s}$ replacing variables by closed values, s.t. $\mathbf{s}(M)$ is Δ-valuable.
- A term M is Δ-solvable if and only if there is a head Δ-context $C[.] \equiv (\lambda\vec{x}.[.])\vec{N}$ such that:

$\vec{x}$ sequentialize variables in $\mathrm{FV}(M)$

$\vec{N} \subseteq \Delta$ and $C[M] =_{\Delta} I$

Γ-Solvability, Γ and Γ-normal forms

- We already showed that the term:

$$(\lambda y.D)(xI)D$$

where $D \equiv \lambda x.xx$
is a Γ-normal form, which is not a Γ-input value, and which is operationally equivalent to DD

Γ-Solvability, Γ and Γ-normal forms

- We already showed that the term:

$$(\lambda y.D)(xI)D$$

where $D \equiv \lambda x.xx$
is a Γ-normal form, which is not a Γ-input value, and which is operationally equivalent to DD

- DD is Γ-unsolvable

Γ-Solvability, Γ and Γ-normal forms

- We already showed that the term:

$$(\lambda y.D)(xI)D$$

where $D \equiv \lambda x.xx$
is a Γ-normal form, which is not a Γ-input value, and which is operationally equivalent to DD

- DD is Γ-unsolvable
- $(\lambda y.D)(xI)D$ is Γ-unsolvable!

Γ-Solvability, Γ and Γ-normal forms

- We already showed that the term:

$$(\lambda y.D)(xI)D$$

where $D \equiv \lambda x.xx$
is a Γ-normal form, which is not a Γ-input value, and which is operationally equivalent to DD

- DD is Γ-unsolvable
- $(\lambda y.D)(xI)D$ is Γ-unsolvable!
- $\lambda x.(\lambda y.D)(xI)D \in \Gamma$!

Γ-Solvability, Γ and Γ-normal forms

- We already showed that the term:

$$(\lambda y.D)(xI)D$$

where $D \equiv \lambda x.xx$
is a Γ-normal form, which is not a Γ-input value, and which is operationally equivalent to DD

- DD is Γ-unsolvable
- $(\lambda y.D)(xI)D$ is Γ-unsolvable!
- $\lambda x.(\lambda y.D)(xI)D \in \Gamma$!
- So neither the set of Γ-normal forms nor the set Γ is not a proper subset of the set of Γ-solvable terms!

Call-by-name, call-by-value relations

Some interesting sets of terms in call-by-value calculi correspond to semantically relevant set of the classical (call-by-name) calculus.

- The set of potentially Γ-valuable terms coincides with the set of lazy Λ-strongly normalizing terms.

Call-by-name, call-by-value relations

Some interesting sets of terms in call-by-value calculi correspond to semantically relevant set of the classical (call-by-name) calculus.

- The set of potentially Γ-valuable terms coincides with the set of lazy Λ-strongly normalizing terms.
- The set of Φ-solvable terms coincides with the set of Λ-strongly normalizing

  (The $\lambda\Phi$-calculus is a not lazy call-by-value calculus.)

Lazy Reductions

Let $\Delta \subseteq \Lambda$ be a set of terms.

- The lazy Δ-reduction ($\rightarrow_{\Delta\ell}$) is the applicative closure of the following rule:

$$(\lambda x.M)N \rightarrow M[N/x] \text{ if and only if } N \in \Delta$$

Lazy Reductions

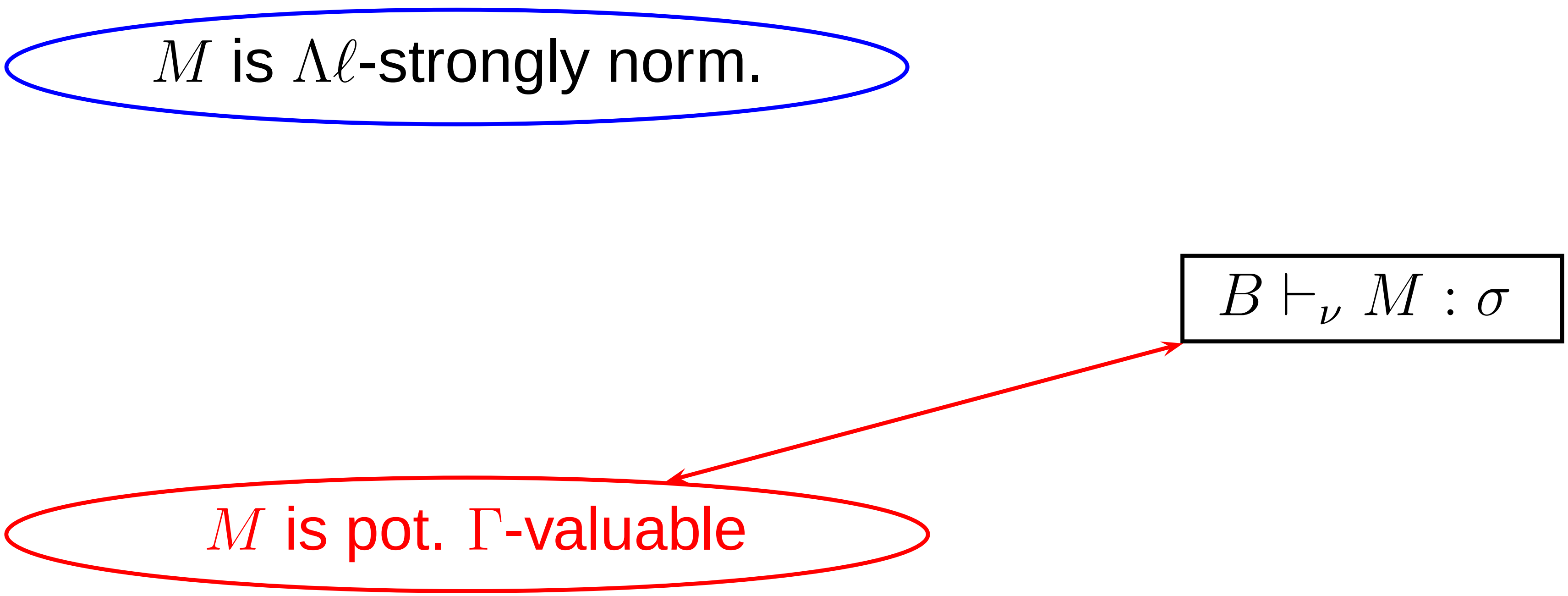
Let $\Delta \subseteq \Lambda$ be a set of terms.

- The lazy Δ-reduction ($\rightarrow_{\Delta\ell}$) is the applicative closure of the following rule:

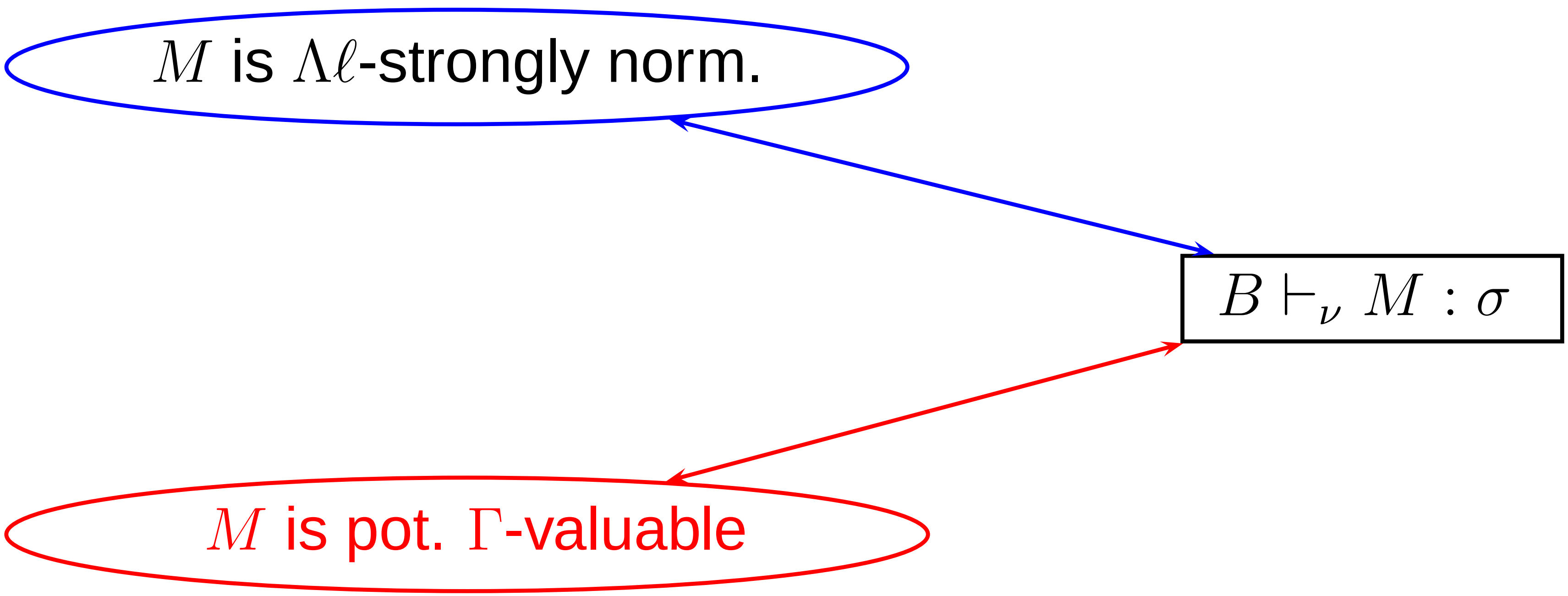
$$(\lambda x.M)N \rightarrow M[N/x] \text{ if and only if } N \in \Delta$$

- Let $I \equiv \lambda x.x$; then $\lambda x.II \rightarrow_{\Lambda} \lambda x.I$
  while $\lambda x.II \not\rightarrow_{\Lambda\ell} \lambda x.I$

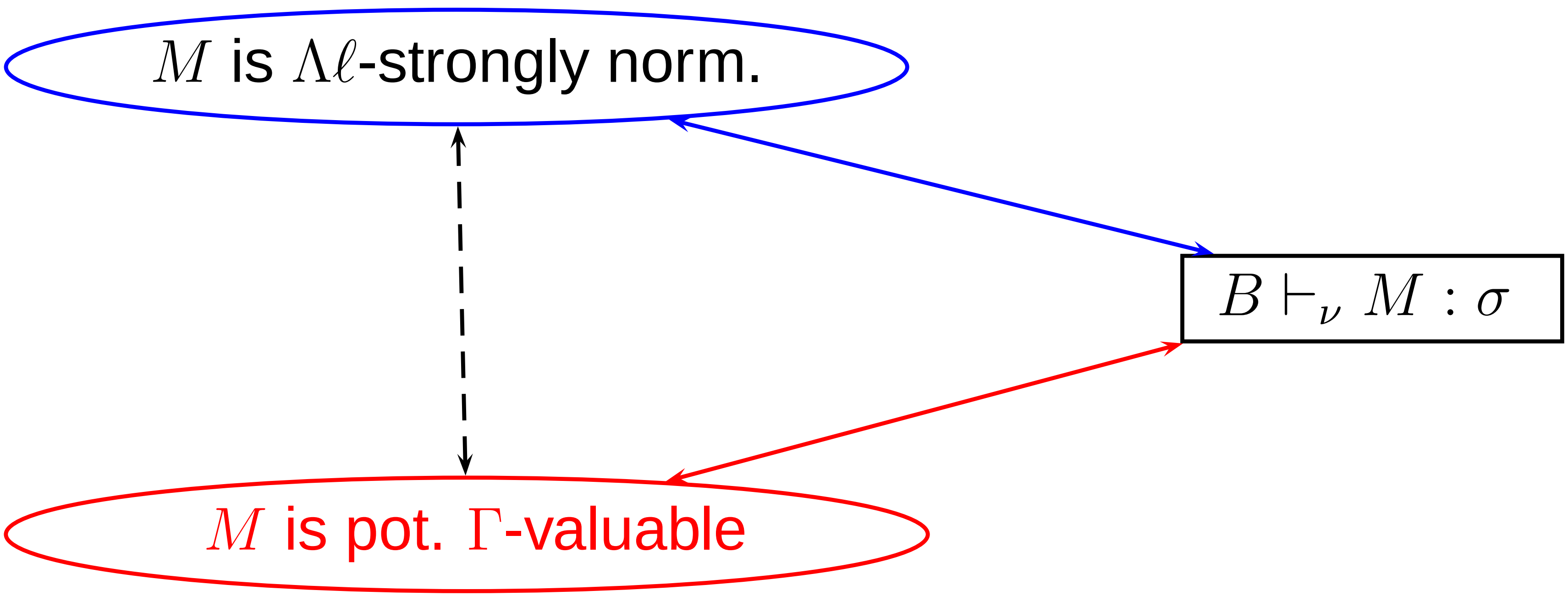
Γ-pot. valuability and $\Lambda\ell$-strongly norm.



Γ-pot. valuability and $\Lambda\ell$-strongly norm.



Γ-pot. valuability and $\Lambda\ell$-strongly norm.



Γ-solvability and $\Lambda\ell$-strong norm.

- The set of Γ-solvable terms is a proper subset of the set of Γ-potentially valuable terms, so it is a proper subset of the set of $\Lambda\ell$-strongly normalizing terms.

Γ-solvability and $\Lambda\ell$-strong norm.

- The set of Γ-solvable terms is a proper subset of the set of Γ-potentially valuable terms, so it is a proper subset of the set of $\Lambda\ell$-strongly normalizing terms.
- The Γ-normal form $(\lambda y.D)(xI)D$ is Γ-unsolvable!

Γ-solvability and $\Lambda\ell$-strong norm.

- The set of Γ-solvable terms is a proper subset of the set of Γ-potentially valuable terms, so it is a proper subset of the set of $\Lambda\ell$-strongly normalizing terms.
- The Γ-normal form $(\lambda y.D)(xI)D$ is Γ-unsolvable!

A non lazy Call-By-Value

- The set SN_Λ of $\Lambda\ell$-strongly normalizing terms is a set of input values.

A non lazy Call-By-Value

- The set SN_Λ of $\Lambda\ell$-strongly normalizing terms is a set of input values.
- The set of SN_Λ-solvable terms coincides with the set of SN_Λ-potentially valuable terms.

A non lazy Call-By-Value

- The set SN_Λ of $\Lambda\ell$-strongly normalizing terms is a set of input values.
- The set of SN_Λ-solvable terms coincides with the set of SN_Λ-potentially valuable terms.
- The set of SN_Λ-solvable terms coincides with SN_Λ.

A non lazy Call-By-Value

- The set SN_Λ of $\Lambda\ell$-strongly normalizing terms is a set of input values.
- The set of SN_Λ-solvable terms coincides with the set of SN_Λ-potentially valuable terms.
- The set of SN_Λ-solvable terms coincides with SN_Λ.
- SN_Λ is not the minimal set of input values satisfying these properties.

A non lazy Call-By-Value

- Υ_i, Φ_i are defined as follows:

$$\Upsilon_0 = \text{Var} \qquad\qquad \Phi_i = \text{Var} \cup (\Upsilon_i)^0$$

$$\begin{aligned}\Upsilon_{i+1} = \text{Var} &\cup \{xM_1...M_n \mid M_k \in \Upsilon_i (1 \leq k \leq n)\} \\ &\cup \{\lambda \vec{x}.M \mid M \in \Upsilon_i\} \\ &\cup \{(\lambda x.P)QM_1...M_n \mid Q \in \Upsilon_i - (\Lambda^0 \cup \text{Var}), \\ &\qquad\qquad P[Q/x]M_1 \ldots M_n \rightarrow^*_{\Phi_i} R \in \Upsilon_i\}\end{aligned}$$

A non lazy Call-By-Value

- Υ_i, Φ_i are defined as follows:

$$\Upsilon_0 = \mathrm{Var} \qquad\qquad \Phi_i = \mathrm{Var} \cup (\Upsilon_i)^0$$

$$\begin{aligned}\Upsilon_{i+1} = \mathrm{Var} & \cup \{xM_1 ... M_n \mid M_k \in \Upsilon_i (1 \leq k \leq n)\} \\ & \cup \{\lambda\vec{x}.M \mid M \in \Upsilon_i\} \\ & \cup \{(\lambda x.P)QM_1 ... M_n \mid Q \in \Upsilon_i - (\Lambda^0 \cup \mathrm{Var}), \\ & \qquad P[Q/x]M_1 \ldots M_n \to^*_{\Phi_i} R \in \Upsilon_i\}\end{aligned}$$

- Let $\Upsilon = \cup_i \Upsilon_i$ and $\Phi = \mathrm{Var} \cup (\Upsilon)^0$

Some Properties of $\lambda\Phi$-calculus

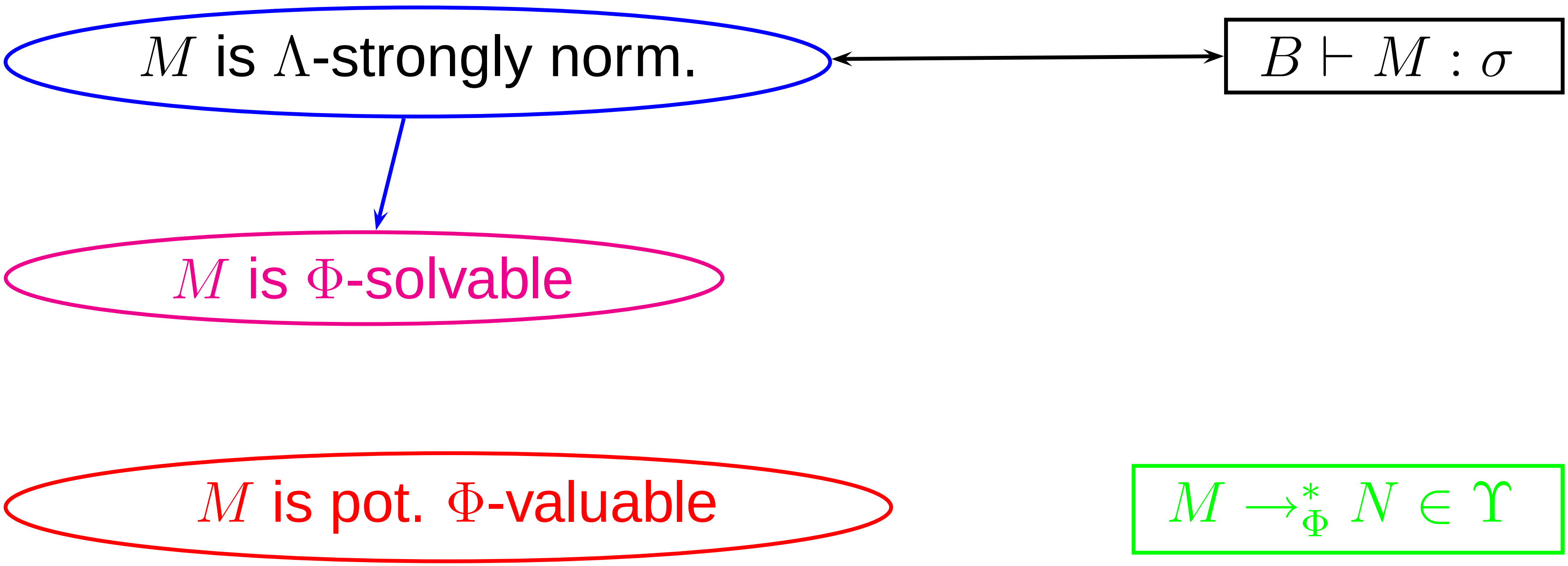
M is Λ-strongly norm. $\longleftrightarrow$ $B \vdash M : \sigma$

M is Φ-solvable

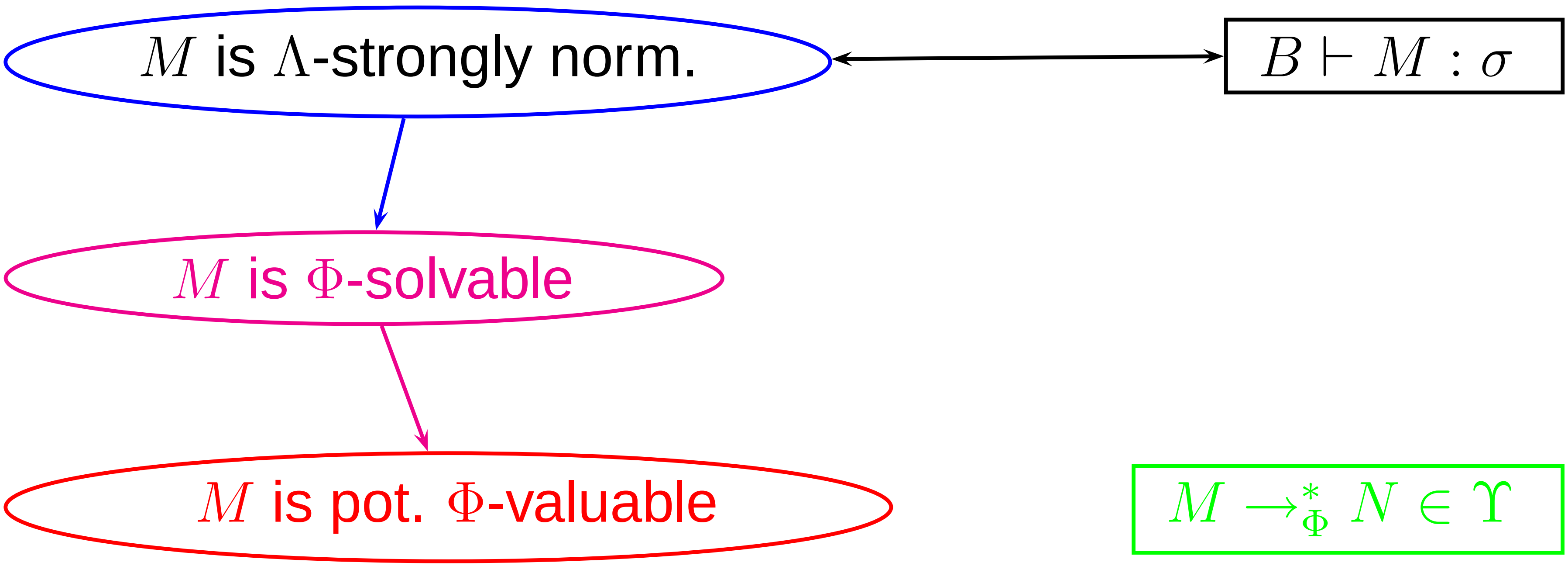
M is pot. Φ-valuable

$M \to_{\Phi}^{*} N \in \Upsilon$

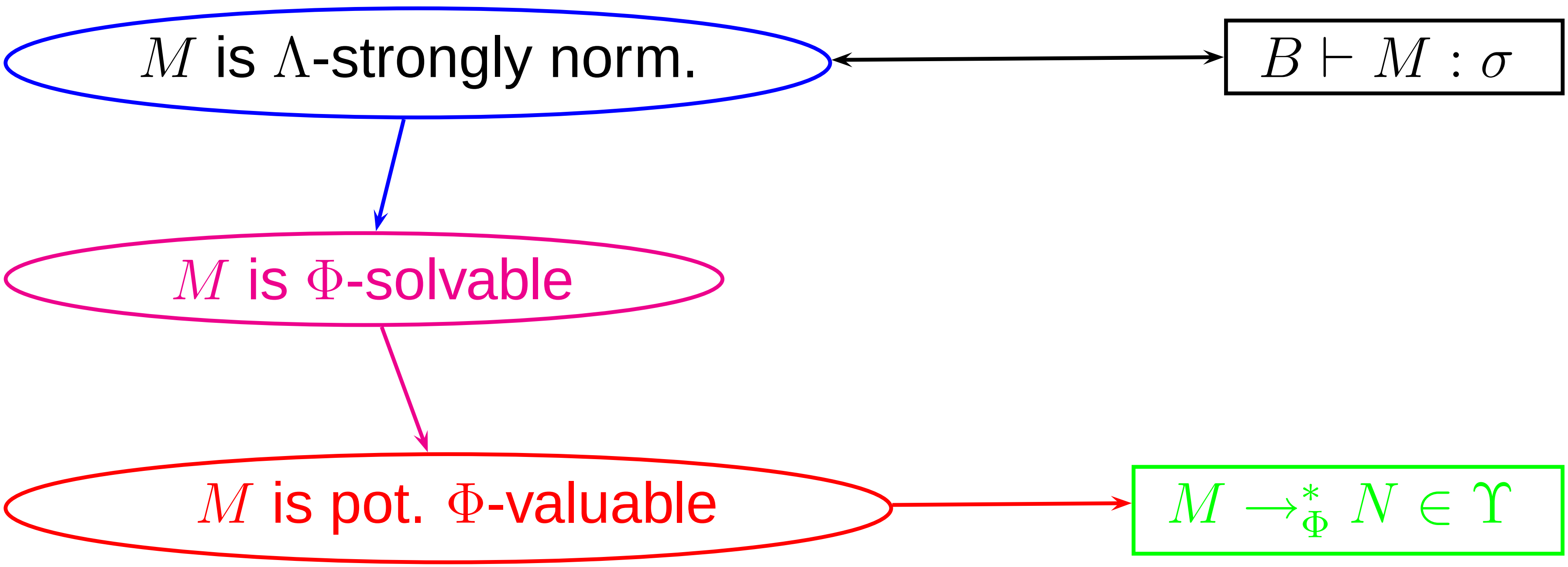
Some Properties of $\lambda\Phi$-calculus



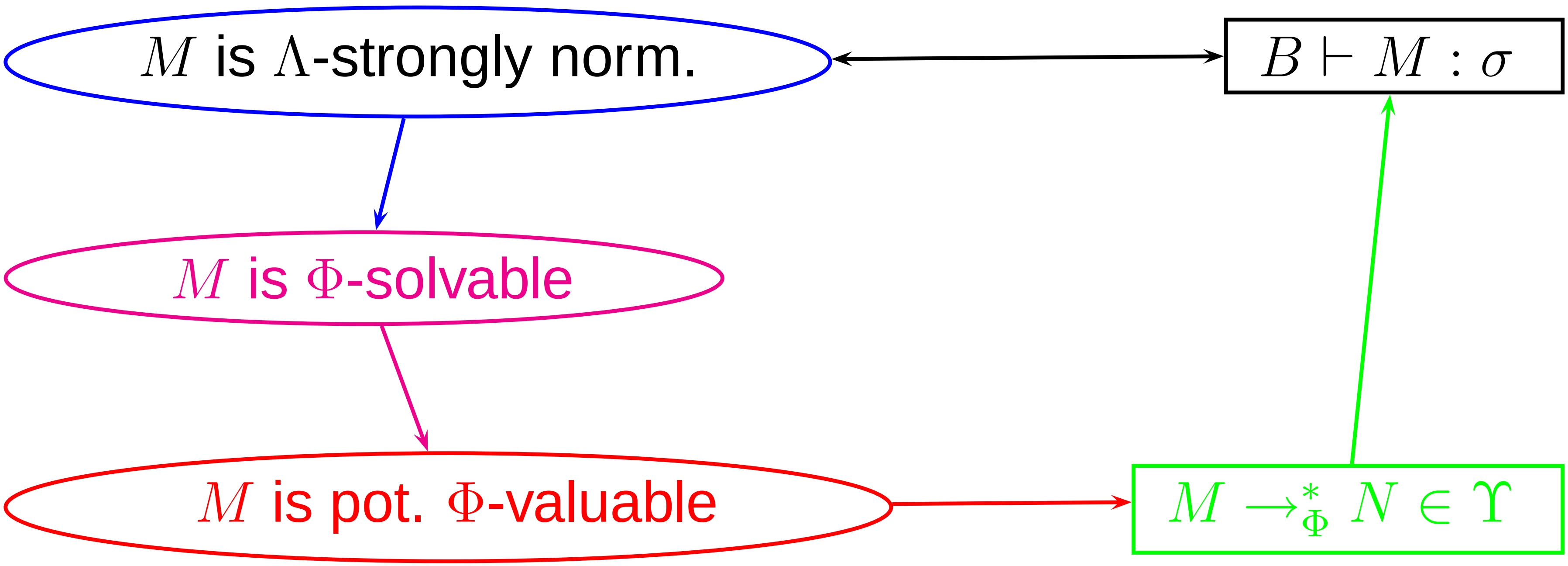
Some Properties of $\lambda\Phi$-calculus



Some Properties of $\lambda\Phi$-calculus



Some Properties of $\lambda\Phi$-calculus



$\lambda\Delta$-model

$\lambda\Delta$-model is a quadruple $< \mathbb{D}, \mathbb{I}, \circ, [\![.]\!] >$ where

- $\mathbb{D}$ is a set
- $\circ$ is an operation on $\mathbb{D}$
- $\mathbb{I}$ is a subset of $\mathbb{D}$
- if $\rho : \mathrm{Var} \rightarrow \mathbb{I}$ then $[\![.]\!]_\rho$ maps terms in $\mathbb{D}$

$\lambda\Delta$-model

$\lambda\Delta$-model is a quadruple $< \mathbb{D}, \mathbb{I}, \circ, [\![.]\!] >$ s.t.:

- $[\![x]\!]_\rho = \rho(x)$

$\lambda\Delta$-model

$\lambda\Delta$-model is a quadruple $< \mathbb{D}, \mathbb{I}, \circ, [\![.]\!] >$ s.t.:

- $[\![x]\!]_\rho = \rho(x)$
- $[\![MN]\!]_\rho = [\![M]\!]_\rho \circ [\![N]\!]_\rho$

$\lambda\Delta$-model

$\lambda\Delta$-model is a quadruple $< \mathbb{D}, \mathbb{I}, \circ, [\![.]\!] >$ s.t.:

- $[\![x]\!]_\rho = \rho(x)$
- $[\![MN]\!]_\rho = [\![M]\!]_\rho \circ [\![N]\!]_\rho$
- $[\![\lambda x.M]\!]_\rho \circ d = [\![M]\!]_{\rho[d/x]}$ if $d \in \mathbb{I}$

$\lambda\Delta$-model

$\lambda\Delta$-model is a quadruple $< \mathbb{D}, \mathbb{I}, \circ, [\![.]\!] >$ s.t.:

- $[\![x]\!]_\rho = \rho(x)$
- $[\![MN]\!]_\rho = [\![M]\!]_\rho \circ [\![N]\!]_\rho$
- $[\![\lambda x.M]\!]_\rho \circ d = [\![M]\!]_{\rho[d/x]}$ if $d \in \mathbb{I}$
- if $[\![M]\!]_{\rho[d/x]} = [\![M']\!]_{\rho'[d/y]}$ for each $d \in \mathbb{I}$,

  then $[\![\lambda x.M]\!]_\rho = [\![\lambda y.M']\!]_{\rho'}$

$\lambda\Delta$-model

$\lambda\Delta$-model is a quadruple $< \mathbb{D}, \mathbb{I}, \circ, [\![.]\!] >$ s.t.:

- $[\![x]\!]_\rho = \rho(x)$
- $[\![MN]\!]_\rho = [\![M]\!]_\rho \circ [\![N]\!]_\rho$
- $[\![\lambda x.M]\!]_\rho \circ d = [\![M]\!]_{\rho[d/x]}$ if $d \in \mathbb{I}$
- if $[\![M]\!]_{\rho[d/x]} = [\![M']\!]_{\rho'[d/y]}$ for each $d \in \mathbb{I}$,

  then $[\![\lambda x.M]\!]_\rho = [\![\lambda y.M']\!]_{\rho'}$
- $M \in \Delta$ implies $\forall \rho.[\![M]\!]_\rho \in \mathbb{I}$

Intersection Types

Let C be a countable set of type-constants, containing at least the constant ω.

Intersection Types

Let C be a countable set of type-constants, containing at least the constant ω.

The set $T(C)$ of types is inductively defined as follows:

$$\sigma ::= \alpha \mid (\sigma \rightarrow \tau) \mid (\sigma \wedge \tau)$$

where $\alpha \in C$.

Intersection Type Theory

A intersection relation $\leq$ is a preorder relation on $T(C)$, closed under the following rules:

$$\frac{}{\sigma \leq \omega}\ (a) \qquad \frac{}{\sigma \leq \sigma}\ (r) \qquad \frac{\sigma \leq \rho,\ \rho \leq \tau}{\sigma \leq \rho}\ (t)$$

$$\frac{}{\sigma \leq \sigma \wedge \sigma}\ (b) \qquad \frac{}{\sigma \wedge \tau \leq \sigma}\ (c) \qquad \frac{}{\sigma \wedge \tau \leq \tau}\ (c')$$

$$\frac{}{(\sigma \to \tau) \wedge (\sigma \to \pi) \leq \sigma \to (\tau \wedge \pi)}\ (d) \qquad \frac{\sigma \leq \sigma',\ \tau \leq \tau'}{\sigma \wedge \tau \leq \sigma' \wedge \tau'}\ (e)$$

$$\frac{\sigma' \leq \sigma,\ \tau \leq \tau'}{\sigma \to \tau \leq \sigma' \to \tau'}\ (f) \qquad \frac{}{\sigma \to \omega \leq \omega \to \omega}\ (g)$$

$\leq$ induce a type theory $\simeq$.

Intersection Type System

A type system ∇ is a triple $< C, \leq_{\nabla}, I(C) >$:

- C is a set of type constants

Intersection Type System

A type system ∇ is a triple $< C, \leq_{\nabla}, I(C) >$:

- C is a set of type constants
- $\leq_{\nabla}$ is an intersection relation on $T(C)$

Intersection Type System

A type system ∇ is a triple $< C, \leq_{\nabla}, I(C) >$:

- C is a set of type constants
- $\leq_{\nabla}$ is an intersection relation on $T(C)$
- $I(C) \subseteq T(C)$ is a set of input types s.t.
  - $I(C)$ is not empty

Intersection Type System

A type system ∇ is a triple $< C, \leq_\nabla, I(C) >$:

- C is a set of type constants
- $\leq_\nabla$ is an intersection relation on $T(C)$
- $I(C) \subseteq T(C)$ is a set of input types s.t.
  - $I(C)$ is not empty
  - $\sigma \in I(C)$ and $\sigma \simeq_\nabla \tau$ imply $\tau \in I(C)$;

Intersection Type System

A type system ∇ is a triple $< C, \leq_\nabla, I(C) >$:

- C is a set of type constants
- $\leq_\nabla$ is an intersection relation on $T(C)$
- $I(C) \subseteq T(C)$ is a set of input types s.t.
  - $I(C)$ is not empty
  - $\sigma \in I(C)$ and $\sigma \simeq_\nabla \tau$ imply $\tau \in I(C)$;
  - $\sigma \in I(C)$ and $\tau \notin I(C)$ imply $\sigma \leq_\nabla \tau$.

Intersection Type Assignment System

Let ∇ be a type system

$$\frac{}{B[\sigma/x] \vdash_\nabla x : \sigma}\ (var) \qquad \frac{}{B \vdash_\nabla M : \omega}\ (\omega)$$

$$\frac{B[\sigma/x] \vdash_\nabla M : \tau}{B \vdash_\nabla \lambda x.M : \sigma \to \tau}\ (\to I) \qquad \frac{\sigma \in I(C) \quad \begin{array}{c} B \vdash_\nabla M : \sigma \to \tau \\ B \vdash_\nabla N : \sigma \end{array}}{B \vdash_\nabla MN : \tau}\ (\to E)$$

$$\frac{B \vdash_\nabla M : \sigma \quad B \vdash_\nabla M : \tau}{B \vdash_\nabla M : \sigma \wedge \tau}\ (\wedge I) \qquad \frac{B \vdash_\nabla M : \sigma \quad \sigma \leq_\nabla \tau}{B \vdash_\nabla M : \tau}\ (\leq_\nabla)$$

$$\frac{B \vdash_\nabla M : \sigma \wedge \tau}{B \vdash_\nabla M : \sigma}\ (\wedge E_l) \qquad \frac{B \vdash_\nabla M : \sigma \wedge \tau}{B \vdash_\nabla M : \tau}\ (\wedge E_r)$$

Intersection Type Assignment System

Let ∇ be a type system

$$\frac{}{B[\sigma/x] \vdash_\nabla x : \sigma}\ (var) \qquad \frac{}{B \vdash_\nabla M : \omega}\ (\omega)$$

$$\frac{B[\sigma/x] \vdash_\nabla M : \tau}{B \vdash_\nabla \lambda x.M : \sigma \to \tau}\ (\to I) \qquad \frac{\boxed{\sigma \in I(C)} \quad B \vdash_\nabla M : \sigma \to \tau \quad B \vdash_\nabla N : \sigma}{B \vdash_\nabla MN : \tau}\ (\to E)$$

$$\frac{B \vdash_\nabla M : \sigma \quad B \vdash_\nabla M : \tau}{B \vdash_\nabla M : \sigma \wedge \tau}\ (\wedge I) \qquad \frac{B \vdash_\nabla M : \sigma \quad \sigma \leq_\nabla \tau}{B \vdash_\nabla M : \tau}\ (\leq_\nabla)$$

$$\frac{B \vdash_\nabla M : \sigma \wedge \tau}{B \vdash_\nabla M : \sigma}\ (\wedge E_l) \qquad \frac{B \vdash_\nabla M : \sigma \wedge \tau}{B \vdash_\nabla M : \tau}\ (\wedge E_r)$$

Filters

Let ∇ be the type system $< C, \leq_{\nabla}, I(C) >$.

Let $\mathcal{F}(\nabla)$ be the set of all filters.

Filters

Let ∇ be the type system $< C, \leq_\nabla, I(C) >$.

Let $\mathcal{F}(\nabla)$ be the set of all filters.

Let $\mathcal{I}(\nabla)$ be the set of input filters containing at least one type belonging to $I(C)$.

Filters

Let ∇ be the type system $< C, \leq_\nabla, I(C) >$.

Let $\mathcal{F}(\nabla)$ be the set of all filters.

Let $\mathcal{I}(\nabla)$ be the set of input filters containing at least one type belonging to $I(C)$.

$$f_1 \circ_\nabla f_2 = \uparrow \{\omega\} \cup \{\tau \mid \sigma \to \tau \in f_1 \text{ and } \sigma \in f_2 \text{ and } \boxed{\sigma \in I(C)}\}.$$

Filters

Let ∇ be the type system $< C, \leq_{\nabla}, I(C) >$.

Let $\mathcal{F}(\nabla)$ be the set of all filters.

Let $\mathcal{I}(\nabla)$ be the set of input filters containing at least one type belonging to $I(C)$.

$$f_1 \circ_{\nabla} f_2 = \uparrow \{\omega\} \cup \{\tau \mid \sigma \to \tau \in f_1 \text{ and } \sigma \in f_2 \text{ and } \boxed{\sigma \in I(C)}\}.$$

$$[\![M]\!]_{\rho}^{\mathcal{F}(\nabla)} = \{\sigma \in T(C) \mid \exists B \propto \rho \text{ such that } B \vdash_{\nabla} M : \sigma\}$$

where $B \propto \rho$ when $B(z) \in \rho(z)$, for all z.

Legal Type Theory

Let ∇ be the type system $< C, \leq_\nabla, I(C) >$.

∇ is legal whenever, for all $\sigma \in I(C)$ and $\tau \not\leq_\nabla \omega$:

$$(\sigma_1 \rightarrow \tau_1) \wedge ... \wedge (\sigma_n \rightarrow \tau_n) \leq_\nabla \sigma \rightarrow \tau$$

Legal Type Theory

Let ∇ be the type system $< C, \leq_\nabla, I(C) >$.

∇ is legal whenever, for all $\sigma \in I(C)$ and $\tau \not\simeq_\nabla \omega$:

$$(\sigma_1 \to \tau_1) \wedge ... \wedge (\sigma_n \to \tau_n) \leq_\nabla \sigma \to \tau$$

implies $\exists \{i_1, ..., i_k\} \subseteq \{1, ..., n\}$ s.t.

$$(\sigma_{i_1} \wedge ... \wedge \sigma_{i_k}) \geq_\nabla \sigma \text{ and } (\tau_{i_1} \wedge ... \wedge \tau_{i_k}) \leq_\nabla \tau.$$

Filters models(Coppo-Dezani)

- Let $\nabla =< C, \leq_{\nabla}, I(C) >$ be legal and such that $M \in \Delta$ imply $\forall \rho \; [\![M]\!]_\rho \in \mathcal{I}(\nabla)$.
  Then $< \mathcal{F}(\nabla), \mathcal{I}(\nabla), \circ_{\nabla}, [\![.]\!]^{\mathcal{F}(\nabla)} >$ is a $\lambda\Delta$-model.

Filters models(Coppo-Dezani)

- Let $\nabla =< C, \leq_{\nabla}, I(C) >$ be legal and such that $M \in \Delta$ imply $\forall\rho\ [\![M]\!]_\rho \in \mathcal{I}(\nabla)$.
  Then $< \mathcal{F}(\nabla), \mathcal{I}(\nabla), \circ_{\nabla}, [\![.]\!]^{\mathcal{F}(\nabla)} >$ is a $\lambda\Delta$-model.
- $M \sqsubseteq_{\mathcal{F}} N$ means that $\forall\rho,\ [\![M]\!]^{\mathcal{F}}_\rho \subseteq [\![N]\!]^{\mathcal{F}}_\rho$

Filters models(Coppo-Dezani)

- Let $\nabla =< C, \leq_\nabla, I(C) >$ be legal and such that $M \in \Delta$ imply $\forall \rho \; [\![M]\!]_\rho \in \mathcal{I}(\nabla)$.
  Then $< \mathcal{F}(\nabla), \mathcal{I}(\nabla), \circ_\nabla, [\![.]\!]^{\mathcal{F}(\nabla)} >$ is a $\lambda\Delta$-model.
- $M \sqsubseteq_{\mathcal{F}} N$ means that $\forall \rho, \; [\![M]\!]^{\mathcal{F}}_\rho \subseteq [\![N]\!]^{\mathcal{F}}_\rho$
- Correctness: $\sqsubseteq_{\mathcal{F}} \subseteq \preceq_{\mathbf{O}}$

Filters models(Coppo-Dezani)

- Let $\nabla =< C, \leq_{\nabla}, I(C) >$ be legal and such that $M \in \Delta$ imply $\forall \rho \; [\![M]\!]_\rho \in \mathcal{I}(\nabla)$. Then $< \mathcal{F}(\nabla), \mathcal{I}(\nabla), \circ_{\nabla}, [\![.]\!]^{\mathcal{F}(\nabla)} >$ is a $\lambda\Delta$-model.
- $M \sqsubseteq_{\mathcal{F}} N$ means that $\forall \rho, \; [\![M]\!]^{\mathcal{F}}_\rho \subseteq [\![N]\!]^{\mathcal{F}}_\rho$
- Correctness: $\sqsubseteq_{\mathcal{F}} \subseteq \preceq_{\mathbf{O}}$
- Completeness: $\preceq_{\mathbf{O}} \subseteq \sqsubseteq_{\mathcal{F}}$

Filters models(Coppo-Dezani)

- Let $\nabla =< C, \leq_{\nabla}, I(C) >$ be legal and such that $M \in \Delta$ imply $\forall\rho\ [\![M]\!]_\rho \in \mathcal{I}(\nabla)$.
  Then $< \mathcal{F}(\nabla), \mathcal{I}(\nabla), \circ_{\nabla}, [\![.]\!]^{\mathcal{F}(\nabla)} >$ is a $\lambda\Delta$-model.
- $M \sqsubseteq_{\mathcal{F}} N$ means that $\forall\rho,\ [\![M]\!]^{\mathcal{F}}_\rho \subseteq [\![N]\!]^{\mathcal{F}}_\rho$
- Correctness: $\sqsubseteq_{\mathcal{F}} \subseteq \preceq_{\mathbf{O}}$
- Completeness: $\preceq_{\mathbf{O}} \subseteq \sqsubseteq_{\mathcal{F}}$
- Full abstraction: $\sqsubseteq_{\mathcal{F}} = \preceq_{\mathbf{O}}$

Using Parametric Models

In literature standard topological models for lazy evaluations of λ-calculus have been considered.

Using Parametric Models

In literature standard topological models for lazy evaluations of λ-calculus have been considered.

Abramsky and Ong (1993)
have studied the call by name parameter passing

Using Parametric Models

In literature standard topological models for lazy evaluations of λ-calculus have been considered.

Abramsky and Ong (1993)
have studied the call by name parameter passing
Egidi, Honsell and Ronchi Della Rocca (1992)
have studied the call by value parameter passing

Using Parametric Models

In literature standard topological models for lazy evaluations of λ-calculus have been considered.

Both models are correct, but not complete!

Using Parametric Models

In literature standard topological models for lazy evaluations of λ-calculus have been considered.

Both models are correct, but not complete!

A pre-order relation on terms, stratified on intersection types which grasps exactly the two operational semantics we want to model.
Such a relation can be used for building
two fully abstract models starting from a logical descriptions of the two previous models.

Using Parametric Models

In literature standard topological models for lazy evaluations of λ-calculus have been considered.

Both models are correct, but not complete!

A pre-order relation on terms, stratified on intersection types which grasps exactly the two operational semantics we want to model.
Such a relation can be used for building
two fully abstract models starting from a logical descriptions of the two previous models.

Lazy Evaluations (Plotkin 1975)

Call by name (said simply lazy)

$$\frac{}{\lambda x.M \Downarrow_{\mathbf{L}} \lambda x.M}\ (lazy) \qquad \frac{P[Q/x]M_1 \ldots M_m \Downarrow_{\mathbf{L}} N}{(\lambda x.P)QM_1 \ldots M_m \Downarrow_{\mathbf{L}} N}\ (head)$$

Call by value

$$\frac{}{\lambda x.M \Downarrow_{\mathbf{V}} \lambda x.M}\ (lazy)$$

$$\frac{Q \Downarrow_{\mathbf{V}} Q' \qquad P[Q'/x]M_1 \ldots M_m \Downarrow_{\mathbf{V}} N}{(\lambda x.P)QM_1 \ldots M_m \Downarrow_{\mathbf{V}} N}\ (head)$$

Operational Semantics

Let $M, N \in \Lambda^0$

$M\Downarrow_{\mathbf{L}}$ if and only if M Λ-reduces to an abstr.

Operational Semantics

Let $M, N \in \Lambda^0$

$M \Downarrow_{\mathbf{V}}$ if and only if M Γ-reduces to an abstr.

Operational Semantics

Let $M, N \in \Lambda^0$

$M \Downarrow_{\mathbf{V}}$ if and only if M Γ-reduces to an abstr.

Let $\mathbf{O} \in \{\mathbf{L}, \mathbf{V}\}$.

Operational Semantics

Let $M, N \in \Lambda^0$

$M \Downarrow_{\mathbf{V}}$ if and only if M Γ-reduces to an abstr.

Let $\mathbf{O} \in \{\mathbf{L}, \mathbf{V}\}$.

$M \preceq_{\mathbf{O}} N$ if and only if,

for all context $C[.]$ such that $C[M], C[N] \in \Lambda^0$,

$C[M] \Downarrow_{\mathbf{O}}$ implies $C[N] \Downarrow_{\mathbf{O}}$.

Operational Semantics

Let $M, N \in \Lambda^0$

$M \Downarrow_{\mathbf{V}}$ if and only if M Γ-reduces to an abstr.

Let $\mathbf{O} \in \{\mathbf{L}, \mathbf{V}\}$.

$M \preceq_{\mathbf{O}} N$ if and only if,

for all context $C[.]$ such that $C[M], C[N] \in \Lambda^0$,

$C[M] \Downarrow_{\mathbf{O}}$ implies $C[N] \Downarrow_{\mathbf{O}}$.

$M \approx_{\mathbf{O}} N$ if and only if $M \preceq_{\mathbf{O}} N$ and $N \preceq_{\mathbf{O}} M$.

$\mathcal{L}$-model (Abramsky-Ong)

- $\angle$ is the type system $< C_\angle, \leq_\angle, I(C_\angle) >$ where $\leq_\angle$ is the least intersection relation, $C_\angle = \{\omega\}$ and $I(C_\angle) = T(C_\angle)$.

$\mathcal{L}$-model (Abramsky-Ong)

- $\angle$ is the type system $< C_\angle, \leq_\angle, I(C_\angle) >$ where $\leq_\angle$ is the least intersection relation, $C_\angle = \{\omega\}$ and $I(C_\angle) = T(C_\angle)$.
- $\mathcal{L}$ is the model $< \mathcal{F}(\angle), \mathcal{F}(\angle), \circ_\angle, [\![.]\!]^{\mathcal{F}(\angle)} >$.
  $\mathcal{L}$ is isomorphic to the topological λ-model, initial solution of: $\mathbb{X} = [\mathbb{X} \to \mathbb{X}]_\perp$

$\mathcal{L}$-model (Abramsky-Ong)

- $\angle$ is the type system $< C_\angle, \leq_\angle, I(C_\angle) >$ where $\leq_\angle$ is the least intersection relation, $C_\angle = \{\omega\}$ and $I(C_\angle) = T(C_\angle)$.
- $\mathcal{L}$ is the model $< \mathcal{F}(\angle), \mathcal{F}(\angle), \circ_\angle, [\![.]\!]^{\mathcal{F}(\angle)} >$.
  $\mathcal{L}$ is isomorphic to the topological λ-model, initial solution of: $\mathbb{X} = [\mathbb{X} \to \mathbb{X}]_\perp$
- $\mathcal{L}$ is correct but not complete with respect to $\preceq_{\mathbf{L}}$.

$\mathcal{V}$-model (Egidi et al.)

- $\surd$ is the type system $< C_{\surd}, \leq_{\surd}, I(C_{\surd}) >$ where $C_{\surd} = \{\omega\}$, $I(C_{\surd}) = \{\sigma \mid \sigma \not\simeq_{\mathcal{V}} \omega\}$ and $\leq_{\surd}$ is the least intersection relation satisfying:

$$\frac{}{(\omega \to \omega) \to \tau \leq_{\surd} \omega \to \tau} \ (v)$$

$\mathcal{V}$-model (Egidi et al.)

- $\surd$ is the type system $< C_{\surd}, \leq_{\surd}, I(C_{\surd}) >$ where $C_{\surd} = \{\omega\}$, $I(C_{\surd}) = \{\sigma \mid \sigma \not\leq_{\mathcal{V}} \omega\}$ and $\leq_{\surd}$ is the least intersection relation satisfying:

$$\frac{}{(\omega \to \omega) \to \tau \leq_{\surd} \omega \to \tau}\ (v)$$

- $\mathcal{V}$ is the model $< \mathcal{F}(\surd), \mathcal{I}(\surd), \circ_{\surd}, [\![.]\!]^{\mathcal{F}(\surd)} >$.

  $\mathcal{V}$ is isomorphic to the topological λ-model, initial solution of: $\mathbb{X} = [\mathbb{X} \to_{\perp} \mathbb{X}]_{\perp}$.

$\mathcal{V}$-model (Egidi et al.)

- $\surd$ is the type system $< C_{\surd}, \leq_{\surd}, I(C_{\surd}) >$ where $C_{\surd} = \{\omega\}$, $I(C_{\surd}) = \{\sigma \mid \sigma \not\leq_{\mathcal{V}} \omega\}$ and $\leq_{\surd}$ is the least intersection relation satisfying:

$$\frac{}{(\omega \to \omega) \to \tau \leq_{\surd} \omega \to \tau} \; (v)$$

- $\mathcal{V}$ is the model $< \mathcal{F}(\surd), \mathcal{I}(\surd), \circ_{\surd}, [\![.]\!]^{\mathcal{F}(\surd)} >$.

  $\mathcal{V}$ is isomorphic to the topological λ-model, initial solution of: $\mathbb{X} = [\mathbb{X} \to_{\perp} \mathbb{X}]_{\perp}$.

- $\mathcal{V}$ is correct but not complete with respect to $\preceq_{\mathbf{V}}$.

$\mathcal{L}$-Characterizations

- $B \vdash_{\angle} M : \omega \to \omega$ if and only if $M \Downarrow_{\mathbf{L}}$.

$\mathcal{L}$-Characterizations

- $B \vdash_{\angle} M : \omega \to \omega$ if and only if $M \Downarrow_{\mathbf{L}}$.
- If $\sigma \not\simeq_{\angle} \omega$ then $\sigma \simeq_{\angle} \sigma_0 \wedge \ldots \wedge \sigma_n$ such that
$$\forall i \leq n,\ \sigma_i \simeq_{\angle} \tau_1^i \to \ldots \to \tau_{m_i}^i \to \omega \to \omega.$$

$\mathcal{L}$-Characterizations

- $B \vdash_{\angle} M : \omega \to \omega$ if and only if $M \Downarrow_{\mathbf{L}}$.
- If $\sigma \not\simeq_{\angle} \omega$ then $\sigma \simeq_{\angle} \sigma_0 \wedge \ldots \wedge \sigma_n$ such that
  $\forall i \leq n$, $\sigma_i \simeq_{\angle} \tau_1^i \to \ldots \to \tau_{m_i}^i \to \omega \to \omega$.
- $\sigma \simeq_{\angle} \omega$ if and only if $\sigma \equiv \underbrace{\omega \wedge \ldots\ldots \wedge \omega}_{n}$.

$\mathcal{V}$-Characterizations

- $B \vdash_{\surd} M : \omega \to \omega$ if and only if $M \Downarrow_{\mathbf{V}}$.
- If $\sigma \not\simeq_{\surd} \omega$ then $\sigma \simeq_{\surd} \sigma_0 \wedge \ldots \wedge \sigma_n$ such that
  $\forall i \leq n$, $\sigma_i \simeq_{\surd} \tau_1^i \to \ldots \to \tau_{m_i}^i \to \omega \to \omega$.
- $\sigma \simeq_{\surd} \omega$ if and only if $\sigma \equiv \underbrace{\omega \wedge \ldots\ldots \wedge \omega}_{n}$.

$\mathcal{L}$-Typed Relation

$\trianglelefteq^{\mathcal{L}}_{\sigma}$ is a relation on Λ^0 defined as follows:

- $M \trianglelefteq^{\mathcal{L}}_{\omega} N$ is true;

$\mathcal{L}$-Typed Relation

$\trianglelefteq_{\sigma}^{\mathcal{L}}$ is a relation on Λ^0 defined as follows:

- $M \trianglelefteq_{\omega}^{\mathcal{L}} N$ is true;
- $M \trianglelefteq_{\sigma\to\tau}^{\mathcal{L}} N$ where $\tau \simeq_{\angle} \omega$, if and only if $B \vdash_{\angle} M : \omega \to \omega$ implies $B \vdash_{\angle} N : \omega \to \omega$;

$\mathcal{L}$-Typed Relation

$\trianglelefteq_{\sigma}^{\mathcal{L}}$ is a relation on Λ^0 defined as follows:

- $M \trianglelefteq_{\omega}^{\mathcal{L}} N$ is true;
- $M \trianglelefteq_{\sigma\to\tau}^{\mathcal{L}} N$ where $\tau \simeq_{\angle} \omega$, if and only if $B \vdash_{\angle} M : \omega \to \omega$ implies $B \vdash_{\angle} N : \omega \to \omega$;
- $M \trianglelefteq_{\sigma\to\tau}^{\mathcal{L}} N$ where $\tau \not\simeq_{\angle} \omega$, if and only if $P \in \Lambda^0$ and $B \vdash_{\angle} P : \sigma$ imply $MP \trianglelefteq_{\tau}^{\mathcal{L}} NP$;

$\mathcal{L}$-Typed Relation

$\trianglelefteq_\sigma^{\mathcal{L}}$ is a relation on Λ^0 defined as follows:

- $M \trianglelefteq_\omega^{\mathcal{L}} N$ is true;
- $M \trianglelefteq_{\sigma\to\tau}^{\mathcal{L}} N$ where $\tau \simeq_{\angle} \omega$, if and only if
  $B \vdash_{\angle} M : \omega \to \omega$ implies $B \vdash_{\angle} N : \omega \to \omega$;
- $M \trianglelefteq_{\sigma\to\tau}^{\mathcal{L}} N$ where $\tau \not\simeq_{\angle} \omega$, if and only if
  $P \in \Lambda^0$ and $B \vdash_{\angle} P : \sigma$ imply $MP \trianglelefteq_\tau^{\mathcal{L}} NP$;
- $M \trianglelefteq_{\sigma\wedge\tau}^{\mathcal{L}} N$ if and only if $M \trianglelefteq_\sigma^{\mathcal{L}} N$ and $M \trianglelefteq_\tau^{\mathcal{L}} N$.

$\mathcal{L}$-Typed Relation

$\trianglelefteq^{\mathcal{L}}_{\sigma}$ is a relation on Λ^0 defined as follows:

- $M \trianglelefteq^{\mathcal{L}}_{\omega} N$ is true;
- $M \trianglelefteq^{\mathcal{L}}_{\sigma\to\tau} N$ where $\tau \simeq_{\angle} \omega$, if and only if
  $B \vdash_{\angle} M : \omega \to \omega$ implies $B \vdash_{\angle} N : \omega \to \omega$;
- $M \trianglelefteq^{\mathcal{L}}_{\sigma\to\tau} N$ where $\tau \not\simeq_{\angle} \omega$, if and only if
  $P \in \Lambda^0$ and $B \vdash_{\angle} P : \sigma$ imply $MP \trianglelefteq^{\mathcal{L}}_{\tau} NP$;
- $M \trianglelefteq^{\mathcal{L}}_{\sigma\wedge\tau} N$ if and only if $M \trianglelefteq^{\mathcal{L}}_{\sigma} N$ and $M \trianglelefteq^{\mathcal{L}}_{\tau} N$.

$M \trianglelefteq^{\mathcal{L}} N$ if and only if $M \trianglelefteq^{\mathcal{L}}_{\sigma} N$, for all σ.

$\mathcal{L}$-Typed Relation

$\trianglelefteq_{\sigma}^{\mathcal{L}}$ is a relation on Λ^0 defined as follows:

- $M \trianglelefteq_{\omega}^{\mathcal{L}} N$ is true;
- $M \trianglelefteq_{\sigma\to\tau}^{\mathcal{L}} N$ where $\tau \simeq_{\angle} \omega$, if and only if $B \vdash_{\angle} M : \omega \to \omega$ implies $B \vdash_{\angle} N : \omega \to \omega$;
- $M \trianglelefteq_{\sigma\to\tau}^{\mathcal{L}} N$ where $\tau \not\simeq_{\angle} \omega$, if and only if $P \in \Lambda^0$ and $B \vdash_{\angle} P : \sigma$ imply $MP \trianglelefteq_{\tau}^{\mathcal{L}} NP$;
- $M \trianglelefteq_{\sigma\wedge\tau}^{\mathcal{L}} N$ if and only if $M \trianglelefteq_{\sigma}^{\mathcal{L}} N$ and $M \trianglelefteq_{\tau}^{\mathcal{L}} N$.

$M \trianglelefteq^{\mathcal{L}} N$ if and only if $M \trianglelefteq_{\sigma}^{\mathcal{L}} N$, for all σ.

$\mathcal{V}$-Typed Relation

$\trianglelefteq_\sigma^V$ is a relation on Λ^0 defined as follows:

- $M \trianglelefteq_\omega^V N$ is true;
- $M \trianglelefteq_{\sigma\to\tau}^V N$ where $\tau \simeq_{\surd} \omega$, if and only if
  $B \vdash_{\surd} M : \omega \to \omega$ implies $B \vdash_{\surd} N : \omega \to \omega$;
- $M \trianglelefteq_{\sigma\to\tau}^V N$ where $\tau \not\simeq_{\surd} \omega$, if and only if
  $P \in \Gamma^0$-val. and $B \vdash_{\surd} P : \sigma$ imply $MP \trianglelefteq_\tau^V NP$;
- $M \trianglelefteq_{\sigma\wedge\tau}^V N$ if and only if $M \trianglelefteq_\sigma^V N$ and $M \trianglelefteq_\tau^V N$.

$M \trianglelefteq^V N$ if and only if $M \trianglelefteq_\sigma^V N$, for all σ.

$\unlhd^{\mathcal{L}}$-Properties

Let $M, N \in \Lambda^0$.

- $\unlhd^{\mathcal{L}}$ is a preorder relation

$\trianglelefteq^{\mathcal{L}}$-Properties

Let $M, N \in \Lambda^0$.

- $\trianglelefteq^{\mathcal{L}}$ is a preorder relation
- $M \sqsubseteq_{\mathcal{L}} N$ implies $M \trianglelefteq^{\mathcal{L}} N$

$\trianglelefteq^{\mathcal{L}}$-Properties

Let $M, N \in \Lambda^0$.

- $\trianglelefteq^{\mathcal{L}}$ is a preorder relation
- $M \sqsubseteq_{\mathcal{L}} N$ implies $M \trianglelefteq^{\mathcal{L}} N$
- $M \trianglelefteq^{\mathcal{L}} N$ if and only if $M \preceq_{\mathbf{L}} N$

$\trianglelefteq^{\mathcal{L}}$-Properties

Let $M, N \in \Lambda^0$.

- $\trianglelefteq^{\mathcal{L}}$ is a preorder relation
- $M \sqsubseteq_{\mathcal{L}} N$ implies $M \trianglelefteq^{\mathcal{L}} N$
- $M \trianglelefteq^{\mathcal{L}} N$ if and only if $M \preceq_{\mathbf{L}} N$

$M \triangleq N$ if and only if $M \trianglelefteq^{\mathcal{L}} N$ and $N \trianglelefteq^{\mathcal{L}} M$.

$\trianglelefteq^V$-Properties

Let $M, N \in \Lambda^0$.

- $\trianglelefteq^V$ is a preorder relation
- $M \sqsubseteq_{\mathcal{V}} N$ implies $M \trianglelefteq^V N$
- $M \trianglelefteq^V N$ if and only if $M \preceq_{\mathbf{V}} N$

$M \triangleq N$ if and only if $M \trianglelefteq^V N$ and $N \trianglelefteq^V M$.

$\mathcal{L}\mathcal{L}$-model

$\triangleq$ induces a equiv. relation on $\mathcal{F}^0(\angle)$, the set of filters being interpretations of closed terms.

- $[f]$ will be the equivalence class of f w.r.t. $\triangleq$, while $\mathcal{F}^0_{\triangleq}$ will be the set of all such classes.

$\mathcal{L}\mathcal{L}$-model

$\triangleq$ induces a equiv. relation on $\mathcal{F}^0(\angle)$, the set of filters being interpretations of closed terms.

- $[f]$ will be the equivalence class of f w.r.t. $\triangleq$, while $\mathcal{F}^0_{\triangleq}$ will be the set of all such classes.
- $\mathcal{I}^0_{\triangleq} = \{[f] \in \mathcal{F}^0_{\triangleq} \mid \exists M \in \Lambda^0$ s.t. $[\![M]\!]^{\mathcal{F}(\angle)}_{\rho} = f\}$.

$\mathcal{L}\mathcal{L}$-model

$\triangleq$ induces a equiv. relation on $\mathcal{F}^0(\angle)$, the set of filters being interpretations of closed terms.

- $[f]$ will be the equivalence class of f w.r.t. $\triangleq$, while $\mathcal{F}^0_{\triangleq}$ will be the set of all such classes.
- $\mathcal{I}^0_{\triangleq} = \{[f] \in \mathcal{F}^0_{\triangleq} \mid \exists M \in \Lambda^0$ s.t. $[\![M]\!]_\rho^{\mathcal{F}(\angle)} = f\}$.
- $[f] \circ_{\triangleq} [g] = [f \circ_{\angle} g]$, for all $[f], [g] \in \mathcal{F}^0_{\triangleq}$.

$\mathcal{LL}$-model

$\triangleq$ induces a equiv. relation on $\mathcal{F}^0(\angle)$, the set of filters being interpretations of closed terms.

- $[f]$ will be the equivalence class of f w.r.t. $\triangleq$, while $\mathcal{F}^0_{\triangleq}$ will be the set of all such classes.
- $\mathcal{I}^0_{\triangleq} = \{[f] \in \mathcal{F}^0_{\triangleq} \mid \exists M \in \Lambda^0$ s.t. $[\![M]\!]^{\mathcal{F}(\angle)}_{\rho} = f\}$.
- $[f] \circ_{\triangleq} [g] = [f \circ_{\angle} g]$, for all $[f], [g] \in \mathcal{F}^0_{\triangleq}$.
- $[\![M]\!]^{\mathcal{LL}}_{\zeta} = [[\![M]\!]^{\mathcal{F}(\angle)}_{\rho}]$, where $\rho(x) \in \zeta(x)$.

$\mathcal{LL}$-model

$\triangleq$ induces a equiv. relation on $\mathcal{F}^0(\angle)$, the set of filters being interpretations of closed terms.

- $[f]$ will be the equivalence class of f w.r.t. $\triangleq$, while $\mathcal{F}^0_{\triangleq}$ will be the set of all such classes.
- $\mathcal{I}^0_{\triangleq} = \{[f] \in \mathcal{F}^0_{\triangleq} \mid \exists M \in \Lambda^0$ s.t. $[\![M]\!]^{\mathcal{F}(\angle)}_{\rho} = f\}$.
- $[f] \circ_{\triangleq} [g] = [f \circ_{\angle} g]$, for all $[f], [g] \in \mathcal{F}^0_{\triangleq}$.
- $[\![M]\!]^{\mathcal{LL}}_{\zeta} = [[\![M]\!]^{\mathcal{F}(\angle)}_{\rho}]$, where $\rho(x) \in \zeta(x)$.
- Let $\mathcal{LL}$ be the quadruple: $< \mathcal{F}^0_{\triangleq}, \mathcal{I}^0_{\triangleq}, \circ, [\![.]\!]^{\mathcal{LL}} >$.

$\mathcal{VV}$-model

$\triangleq$ induces a equiv. relation on $\mathcal{F}^0(\surd)$, the set of filters being interpretations of closed terms.

- $[f]$ will be the equivalence class of f w.r.t. $\triangleq$, while $\mathcal{F}^0_{\triangleq}$ will be the set of all such classes.
- $\mathcal{I}^0_{\triangleq} = \{[f] \in \mathcal{F}^0_{\triangleq} \mid \exists M \in \Gamma^0$ s.t. $[\![M]\!]^{\mathcal{F}(\surd)}_{\rho} = f\}$.
- $[f] \circ_{\triangleq} [g] = [f \circ_{\surd} g]$, for all $[f], [g] \in \mathcal{F}^0_{\triangleq}$.
- $[\![M]\!]^{\mathcal{VV}}_{\zeta} = [[\![M]\!]^{\mathcal{F}(\surd)}_{\rho}]$, where $\rho(x) \in \zeta(x)$.
- Let $\mathcal{VV}$ be the quadruple: $< \mathcal{F}^0_{\triangleq}, \mathcal{I}^0_{\triangleq}, \circ, [\![.]\!]^{\mathcal{VV}} >$.

Full Abstraction

- $\mathcal{LL}$ is a $\lambda\Lambda$-model.
- $\mathcal{VV}$ is a $\lambda\Gamma$-model.

Full Abstraction

- $\mathcal{LL}$ is a $\lambda\Lambda$-model.
- $\mathcal{VV}$ is a $\lambda\Gamma$-model.
- The model $\mathcal{LL}$ is fully abstract with respect to the **L**-operational semantics.
- The model $\mathcal{VV}$ is fully abstract with respect to the **V**-operational semantics.