An interpretation of system F through bar recursion

Valentin Blot

University of Bath

research funded by the UK EPSRC

Realizability interpretations of $PA2$

- Second-order arithmetic ($PA2$):
  - Quantification on $\mathbb{N}$: $\forall n$
  - Quantification on $\mathcal{P}(\mathbb{N})$: $\forall X$
  - Induction: $\forall X\,(X(0) \Rightarrow \forall n\,(X(n) \Rightarrow X(n+1)) \Rightarrow \forall n\, X(n))$
  - Comprehension: $\exists X \forall n\,(A[n] \Leftrightarrow X(n))$

Realizability interpretations of $PA2$

- Second-order arithmetic ($PA2$):
  - Quantification on $\mathbb{N}$: $\forall n$
  - Quantification on $\mathcal{P}(\mathbb{N})$: $\forall X$
  - Induction: $\forall X\,(X(0) \Rightarrow \forall n\,(X(n) \Rightarrow X(n+1)) \Rightarrow \forall n\,X(n))$
  - Comprehension: $\exists X \forall n\,(A[n] \Leftrightarrow X(n))$
- Realizability for $PA2$
  - in polymorphic λ-calculus:
    - Krivine realizability
    - $\langle \lambda x.x, \lambda x.x \rangle \Vdash \exists X \forall n\,(A[n] \Leftrightarrow X(n))$

Realizability interpretations of $PA2$

- Second-order arithmetic ($PA2$):
  - Quantification on $\mathbb{N}$: $\forall n$
  - Quantification on $\mathcal{P}(\mathbb{N})$: $\forall X$
  - Induction: $\forall X\, (X\,(0) \Rightarrow \forall n\, (X\,(n) \Rightarrow X\,(n+1)) \Rightarrow \forall n\, X\,(n))$
  - Comprehension: $\exists X \forall n\, (A\,[n] \Leftrightarrow X\,(n))$

- Realizability for $PA2$
  - in polymorphic λ-calculus:
    - Krivine realizability
    - $\langle \lambda x.x, \lambda x.x \rangle \Vdash \exists X \forall n\, (A\,[n] \Leftrightarrow X\,(n))$
  - in system T + bar recursion (simply-typed)
    - Spector, Kohlenbach, Berger-Oliva, Berardi-Bezem-Coquand
    - $\mathtt{brec} \Vdash \forall n\, \exists b\, (A\,[n] \Leftrightarrow b) \Rightarrow \exists X\, \forall n\, (A\,[n] \Leftrightarrow X\,(n))$
    - $\Vdash \forall n\, \exists b\, (A\,[n] \Leftrightarrow b)$

Weak head normalization of system F in PA2

Definition (Weak head reduction)

$$(\lambda x.M)\, N\, P_0\, \ldots\, P_{n-1} \succ M\,[N/x]\; P_0\, \ldots\, P_{n-1}$$

Weak head normalization of system F in PA2

Definition (Weak head reduction)

$$(\lambda x.M)\,N\,P_0\,\ldots\,P_{n-1} \succ M\,[N/x]\;P_0\,\ldots\,P_{n-1}$$

Theorem (Weak head normalization of system F)

If $M : T$ in system F then weak head reduction terminates on M

Weak head normalization of system F in PA2

Definition (Weak head reduction)

$$(\lambda x.M)\, N\, P_0\, \ldots\, P_{n-1} \succ M\,[N/x]\; P_0\, \ldots\, P_{n-1}$$

Theorem (Weak head normalization of system F)

If $M : T$ in system F then weak head reduction terminates on M

- Reducibility candidates (sets of λ-terms with some properties)
- Not formalizable in $PA2$ (Gödel's incompleteness)
- But for each $M : T$ there is a proof in $PA2$ that M normalizes
- Indeed, f provably total in $PA2$ iff f representable in F

Weak head normalization of system F in PA2

Definition (Weak head reduction)

$$(\lambda x.M)\, N\, P_0\, \ldots\, P_{n-1} \succ M\,[N/x]\; P_0\, \ldots\, P_{n-1}$$

Theorem (Weak head normalization of system F)

If $M : T$ in system F then weak head reduction terminates on M

- Reducibility candidates (sets of λ-terms with some properties)
- Not formalizable in $PA2$ (Gödel's incompleteness)
- But for each $M : T$ there is a proof in $PA2$ that M normalizes
- Indeed, f provably total in $PA2$ iff f representable in F

The translation of $M : T$ is the bar recursive realizability interpretation of its normalization proof

Outline

A logic for λ-terms (bye bye Gödelitis)

A simply-typed total programming language with bar recusion

A realizability model for our logic

The realizability interpretation of normalization of $M : T$

The translation of $M : T$

A logic for λ-terms (bye bye Gödelitis)

Terms

Multi-sorted first-order logic

- Natural numbers: m
- λ-terms (de Bruijn indices): M
- Applicative contexts (stacks of terms): Π
- Sets of λ-terms: X
- Booleans: Φ

Terms

Multi-sorted first-order logic

- Natural numbers: m
- λ-terms (de Bruijn indices): M
- Applicative contexts (stacks of terms): Π
- Sets of λ-terms: X
- Booleans: Φ

$$
\begin{array}{rcl}
m & ::= & i \mid 0 \mid S\, m \qquad M \;::=\; t \mid \underline{m} \mid \lambda.M \mid M\,\Pi \mid M\,[m \mapsto \Pi] \\
\Pi & ::= & \pi \mid \langle\rangle \mid \langle \Pi, M \rangle \qquad X \qquad \Phi \;::=\; b \mid tt \mid ff \mid M \in X
\end{array}
$$

i, t, π, X and b range over countable sets of sorted variables

Substitutions

Substitutions are part of the language because:

Substitutions

Substitutions are part of the language because:

- A condition for X to be a reducibility candidate is:

$$\forall t \, \forall u \, \forall \pi \, (t \, [0 \mapsto u] \; \pi \in X \implies (\lambda.t) \; u \, \pi \in X)$$

Substitutions

Substitutions are part of the language because:

- A condition for X to be a reducibility candidate is:

$$\forall t\, \forall u\, \forall \pi\, (t\,[0 \mapsto u]\ \pi \in X \Longrightarrow (\lambda.t)\ u\,\pi \in X)$$

- The induction hypothesis of the normalization theorem is:

$$T_{n-1}, \ldots, T_0 \vdash M : U \\ \Longrightarrow \forall t_i \in [T_i]\,, M\,[0 \mapsto \langle t_0, \ldots, t_{n-1} \rangle] \in [U]$$

Substitutions

Substitutions are part of the language because:

- A condition for X to be a reducibility candidate is:

$$\forall t\, \forall u\, \forall \pi\, (t\,[0 \mapsto u]\ \pi \in X \Longrightarrow (\lambda.t)\ u\, \pi \in X)$$

- The induction hypothesis of the normalization theorem is:

$$T_{n-1}, \ldots, T_0 \vdash M : U$$
$$\Longrightarrow \forall t_i \in [T_i]\,, M\,[0 \mapsto \langle t_0, \ldots, t_{n-1} \rangle] \in [U]$$

$M\,[m \mapsto \langle M_0, \ldots, M_{n-1} \rangle]$ replaces variables:

$$\underline{0}, \quad \ldots, \quad \underline{m-1}, \quad \underline{m}, \quad \ldots, \quad \underline{m+n-1}, \quad \underline{m+n}, \quad \ldots$$

with terms:

$$\underline{0}, \quad \ldots, \quad \underline{m-1}, \quad M_0, \quad \ldots, \quad M_{n-1}, \quad \underline{m}, \quad \ldots$$

Atomic predicates

$$P \quad ::= \quad \Phi \mid M{\downarrow}^{m} \mid (\!| m |\!) \mid (\!| M |\!) \mid (\!| \Pi |\!)$$

Atomic predicates

$$P \quad ::= \quad \Phi \mid M{\downarrow}^m \mid (\!|m|\!) \mid (\!|M|\!) \mid (\!|\Pi|\!)$$

- Φ means "$\Phi = tt$"
- $M{\downarrow}^m$ means that weak head reduction terminates on M in at most m steps

Atomic predicates

$$P \quad ::= \quad \Phi \mid M{\downarrow}^m \mid (\!| m |\!) \mid (\!| M |\!) \mid (\!| \Pi |\!)$$

- Φ means "$\Phi = tt$"
- $M{\downarrow}^m$ means that weak head reduction terminates on M in at most m steps
- $(\!| \_ |\!)$ are relativization predicates: their unique realizer is their value (I will come back to this)
- no $(\!| X |\!)$ or $(\!| \Phi |\!)$: sets and booleans never need to be relativized

Formulas

$$A, B \quad ::= \quad P \mid A \Rightarrow B \mid A \wedge B \mid \forall i\, A \mid \forall t\, A \mid \forall \pi\, A \mid \forall X\, A \mid \forall b\, A$$

Formulas

$$A, B \quad ::= \quad P \mid A \Rightarrow B \mid A \wedge B \mid \forall i\, A \mid \forall t\, A \mid \forall \pi\, A \mid \forall X\, A \mid \forall b\, A$$

- Negation defined as: $\neg A \triangleq A \Rightarrow \mathit{ff}$
- Existentials encoded as: $\exists i\, A \triangleq \neg \forall i\, \neg A$, same for t, π, X, b

Formulas

$$A, B \ ::= \ P \mid A \Rightarrow B \mid A \wedge B \mid \forall i\, A \mid \forall t\, A \mid \forall \pi\, A \mid \forall X\, A \mid \forall b\, A$$

- Negation defined as: $\neg A \triangleq A \Rightarrow \mathit{ff}$
- Existentials encoded as: $\exists i\, A \triangleq \neg \forall i\, \neg A$, same for t, π, X, b
- Relativized quantifications defined as: $\forall^r i\, A \triangleq \forall i\, (\llparenthesis i \rrparenthesis \Rightarrow A)$ and $\exists^r i\, A \triangleq \neg \forall^r i\, \neg A$, same for t, π
- A realizer of $\forall^r i\, A$ can depend on i, a realizer of $\forall i\, A$ cannot

Formulas

$$A, B \quad ::= \quad P \mid A \Rightarrow B \mid A \wedge B \mid \forall i\, A \mid \forall t\, A \mid \forall \pi\, A \mid \forall X\, A \mid \forall b\, A$$

- Negation defined as: $\neg A \triangleq A \Rightarrow \textit{ff}$
- Existentials encoded as: $\exists i\, A \triangleq \neg \forall i\, \neg A$, same for t, π, X, b
- Relativized quantifications defined as: $\forall^r i\, A \triangleq \forall i\, (\llparenthesis i \rrparenthesis \Rightarrow A)$ and $\exists^r i\, A \triangleq \neg \forall^r i\, \neg A$, same for t, π
- A realizer of $\forall^r i\, A$ can depend on i, a realizer of $\forall i\, A$ cannot
- Normalization defined as: $M{\downarrow} \triangleq \exists^r i\, M{\downarrow}^i$

Weak head normalization, formally (1)

If $A\,(t)$ is a formula with free variable t, define:

$$\mathcal{R}ed\mathcal{C}and\,(A) \stackrel{\Delta}{=} (\forall^r \pi\, A\,(\underline{0}\,\pi) \wedge \forall^r t\,(A\,(t) \Rightarrow t\!\downarrow))$$
$$\wedge \forall^r t\, \forall^r u\, \forall^r \pi\,(A\,(t\,[0 \mapsto \langle u \rangle]\ \pi) \Rightarrow A\,((\lambda.t)\ \langle u \rangle\ \pi))$$

Weak head normalization, formally (1)

If $A\,(t)$ is a formula with free variable t, define:

$$\mathcal{R}ed\mathcal{C}and\,(A) \stackrel{\Delta}{=} (\forall^r \pi\, A\,(\underline{0}\,\pi) \wedge \forall^r t\,(A\,(t) \Rightarrow t\!\downarrow))$$
$$\wedge\, \forall^r t\, \forall^r u\, \forall^r \pi\, (A\,(t\,[0 \mapsto \langle u\rangle]\;\pi) \Rightarrow A\,((\lambda.t)\;\langle u\rangle\;\pi))$$

If T type of system F built from variables X of the logic, define $RC_T\,(t)$ with free variables $\vec{X}$ and t:

$$RC_X\,(t) \stackrel{\Delta}{=} t \in X \qquad RC_{T\to U}\,(t) \stackrel{\Delta}{=} \forall^r u\,(RC_T\,(u) \Rightarrow RC_U\,(t\,u))$$
$$RC_{\forall X\, T}\,(t) \stackrel{\Delta}{=} \forall X\,\left(\mathcal{R}ed\mathcal{C}and\left(\overline{X}\right) \Rightarrow RC_T\,(t)\right)$$

where $\overline{X}\,(t) \stackrel{\Delta}{=} t \in X$. $RC_T\,(t)$ is what we wrote $t \in [T]$ earlier

Weak head normalization, formally (2)

The proof goes in three steps:

Weak head normalization, formally (2)

The proof goes in three steps:

- $\mathcal{R}ed\mathcal{C}and\,(\Downarrow)$ where $\Downarrow(t) \triangleq t{\downarrow}$

Weak head normalization, formally (2)

The proof goes in three steps:

- $\mathcal{R}ed\mathcal{C}and\,(\Downarrow)$ where $\Downarrow(t) \triangleq t\downarrow$
- If T type of F with $\mathrm{FV}(T) = \{X_0, \ldots, X_{n-1}\}$ then:

$$\forall X_0(\mathcal{R}ed\mathcal{C}and\left(\overline{X_0}\right) \Rightarrow \ldots \Rightarrow \forall X_{n-1}(\mathcal{R}ed\mathcal{C}and\left(\overline{X_{n-1}}\right) \Rightarrow \mathcal{R}ed\mathcal{C}and\,(RC_T)) \ldots)$$

Weak head normalization, formally (2)

The proof goes in three steps:

- $\mathcal{R}ed\mathcal{C}and\,(\Downarrow)$ where $\Downarrow(t) \stackrel{\Delta}{=} t\downarrow$
- If T type of F with $\mathrm{FV}\,(T) = \{X_0, \ldots, X_{n-1}\}$ then:

$$\forall X_0(\mathcal{R}ed\mathcal{C}and\,(\overline{X_0}) \Rightarrow \ldots \Rightarrow \forall X_{n-1}(\mathcal{R}ed\mathcal{C}and\,(\overline{X_{n-1}}) \Rightarrow \mathcal{R}ed\mathcal{C}and\,(RC_T))\ldots)$$

- If $\mathrm{FV}\,(T_0, \ldots, T_{m-1}, U) \subseteq \{X_0, \ldots, X_{n-1}\}$ and $T_{m-1}, \ldots, T_0 \vdash M : U$ typing derivation in F then:

$$\forall X_0(\mathcal{R}ed\mathcal{C}and\,(\overline{X_0}) \Rightarrow \ldots \Rightarrow \forall X_{n-1}(\mathcal{R}ed\mathcal{C}and\,(\overline{X_{n-1}}) \Rightarrow \forall^r t_{m-1}(RC_{T_{m-1}}\,(t_{m-1}) \Rightarrow \ldots \Rightarrow \forall^r t_0(RC_{T_0}\,(t_0) \Rightarrow RC_U\,(M\,[0 \mapsto \langle t_0, \ldots, t_{m-1}\rangle]))\ldots))\ldots)$$

Weak head normalization, formally (3)

In particular if M is a closed term of closed type T in F then:

$$RC_T\,(M) \text{ and } \mathcal{R}ed\mathcal{C}and\,(RC_T)$$

Weak head normalization, formally (3)

In particular if M is a closed term of closed type T in F then:

$$RC_T(M) \text{ and } \mathcal{R}ed\mathcal{C}and(RC_T)$$

therefore by definition of $\mathcal{R}ed\mathcal{C}and(\_)$ we obtain:

$$M\downarrow \text{ i.e. } \exists^r i\, M\downarrow^i$$

Weak head normalization, formally (3)

In particular if M is a closed term of closed type T in F then:

$$RC_T(M) \text{ and } \mathcal{R}ed\mathcal{C}and\,(RC_T)$$

therefore by definition of $\mathcal{R}ed\mathcal{C}and\,(\_)$ we obtain:

$$M\!\downarrow \text{ i.e. } \exists^r i\, M\!\downarrow^i$$

Which is a Σ^0_1 formula

Weak head normalization, formally (3)

In particular if M is a closed term of closed type T in F then:

$$RC_T(M) \text{ and } \mathcal{R}ed\mathcal{C}and(RC_T)$$

therefore by definition of $\mathcal{R}ed\mathcal{C}and(\_)$ we obtain:

$$M{\downarrow} \text{ i.e. } \exists^r i\, M{\downarrow}^i$$

Which is a Σ^0_1 formula

$\rightsquigarrow$ classical extraction will give some $n \in \mathbb{N}$ such that M reaches a weak head normal form in at most n steps!

Weak head normalization, formally (3)

In particular if M is a closed term of closed type T in F then:

$$RC_T\,(M) \text{ and } \mathcal{R}ed\mathcal{C}and\,(RC_T)$$

therefore by definition of $\mathcal{R}ed\mathcal{C}and\,(\_)$ we obtain:

$$M\downarrow \text{ i.e. } \exists^r i\, M\downarrow^i$$

Which is a Σ^0_1 formula

$\rightsquigarrow$ classical extraction will give some $n \in \mathbb{N}$ such that M reaches a weak head normal form in at most n steps!

Then it is straightforward to compute this normal form with primitive recursion

A simply-typed total programming language with bar recusion

Simply-typed λ-calculus with products

Simple types:

$$\sigma, \tau \quad ::= \quad \kappa \mid \top \mid \sigma \to \tau \mid \sigma \times \tau$$

where κ ranges over a set of base types

Simply-typed λ-calculus with products

Simple types:

$$\sigma, \tau \quad ::= \quad \kappa \mid \top \mid \sigma \to \tau \mid \sigma \times \tau$$

where κ ranges over a set of base types

Typing rules:

$$\frac{}{\Gamma, x : \sigma \vdash x : \sigma} \qquad \frac{}{\Gamma \vdash \mathsf{c} : \sigma}\ (\mathsf{c}{:}\sigma) \in \mathcal{C}st$$

$$\frac{\Gamma, x : \sigma \vdash \mathtt{M} : \tau}{\Gamma \vdash \lambda x.\mathtt{M} : \sigma \to \tau} \qquad \frac{\Gamma \vdash \mathtt{M} : \sigma \to \tau \qquad \Gamma \vdash \mathtt{N} : \sigma}{\Gamma \vdash \mathtt{M}\,\mathtt{N} : \tau}$$

$$\frac{\Gamma \vdash \mathtt{M} : \sigma \qquad \Gamma \vdash \mathtt{N} : \tau}{\Gamma \vdash \langle \mathtt{M}, \mathtt{N} \rangle : \sigma \times \tau} \qquad \frac{}{\Gamma \vdash * : \top}$$

$$\frac{\Gamma \vdash \mathtt{M} : \sigma \times \tau}{\Gamma \vdash \mathtt{p}_1\,\mathtt{M} : \sigma} \qquad \frac{\Gamma \vdash \mathtt{M} : \sigma \times \tau}{\Gamma \vdash \mathtt{p}_2\,\mathtt{M} : \tau}$$

where $\mathcal{C}st$ is a set of typed constants

System ΛT

- 3 base types:
  - ι type of natural numbers
  - λ type of λ-terms
  - $\lambda^{\diamond}$ type of finite lists of λ-terms

System ΛT

- 3 base types:
  - ι type of natural numbers
  - λ type of λ-terms
  - $\lambda^\diamond$ type of finite lists of λ-terms
- $\mathcal{C}st$ consists of constructors and iterators on these:
  - for ι: $\mathtt{z} : \iota \quad \mathtt{s} : \iota \to \iota \quad \mathtt{it}_\iota : \sigma \to (\sigma \to \sigma) \to \iota \to \sigma$
  - for λ: $\mathtt{var} : \iota \to \lambda \quad \mathtt{abs} : \lambda \to \lambda \quad \mathtt{app} : \lambda \to \lambda \to \lambda$

    $$\mathtt{it}_\lambda : (\iota \to \sigma) \to (\sigma \to \sigma) \to (\sigma \to \sigma \to \sigma) \to \lambda \to \sigma$$
  - for $\lambda^\diamond$: $\mathtt{nil} : \lambda^\diamond \quad \mathtt{cons} : \lambda^\diamond \to \lambda \to \lambda^\diamond$

    $$\mathtt{it}_{\lambda^\diamond} : \sigma \to (\sigma \to \lambda \to \sigma) \to \lambda^\diamond \to \sigma$$

System ΛT

- 3 base types:
  - ι type of natural numbers
  - λ type of λ-terms
  - $\lambda^\diamond$ type of finite lists of λ-terms
- $\mathcal{C}st$ consists of constructors and iterators on these:
  - for ι: $\mathtt{z} : \iota$ $\quad \mathtt{s} : \iota \to \iota$ $\quad \mathtt{it}_\iota : \sigma \to (\sigma \to \sigma) \to \iota \to \sigma$
  - for λ: $\mathtt{var} : \iota \to \lambda$ $\quad \mathtt{abs} : \lambda \to \lambda$ $\quad \mathtt{app} : \lambda \to \lambda \to \lambda$

    $$\mathtt{it}_\lambda : (\iota \to \sigma) \to (\sigma \to \sigma) \to (\sigma \to \sigma \to \sigma) \to \lambda \to \sigma$$
  - for $\lambda^\diamond$: $\mathtt{nil} : \lambda^\diamond$ $\quad \mathtt{cons} : \lambda^\diamond \to \lambda \to \lambda^\diamond$

    $$\mathtt{it}_{\lambda^\diamond} : \sigma \to (\sigma \to \lambda \to \sigma) \to \lambda^\diamond \to \sigma$$
- Easy to define:
  - $\mathsf{app}^\diamond$ s.t.:

    $$\mathsf{app}^\diamond\, \mathtt{M}\, \langle \mathtt{N}_0 \ldots \mathtt{N}_{n-1} \rangle \rightsquigarrow^* \mathsf{app}\,(\ldots(\mathsf{app}\,\mathtt{M}\,\mathtt{P}_0)\ldots)\,\mathtt{P}_{n-1}$$

    where $\langle \mathtt{N}_0 \ldots \mathtt{N}_{n-1} \rangle \stackrel{\Delta}{=} \mathtt{cons}\,(\ldots(\mathtt{cons}\,\mathtt{nil}\,\mathtt{N}_0)\ldots)\,\mathtt{N}_{n-1}$
    and $\mathtt{N}_i \rightsquigarrow^* \mathtt{P}_i$
  - $\mathtt{M}\,[\mathtt{N} \mapsto \mathtt{P}]$ for $\mathtt{M} : \lambda$, $\mathtt{N} : \iota$, $\mathtt{P} : \lambda$ implementing substitution
  - eq s.t. $\mathsf{eq}\,\mathtt{M}\,\mathtt{N} \rightsquigarrow^* \mathtt{z}$ iff $\mathtt{M} \rightsquigarrow^* \mathtt{P}$ and $\mathtt{N} \rightsquigarrow^* \mathtt{P}$ for some $\mathtt{P}$

Preliminaries for bar recursion: observable partial functions

- Type of observable partial functions on λ:

$$\sigma^{\dagger} \triangleq \lambda \to \iota \times \sigma$$

  - $\mathtt{p}_1\,(\mathtt{M}\,\mathtt{N}) \leadsto^* \mathtt{z}$ iff $\mathtt{M} : \sigma^{\dagger}$ defined in $\mathtt{N} : \lambda$ with value $\mathtt{p}_2\,(\mathtt{M}\,\mathtt{N})$

Preliminaries for bar recursion: observable partial functions

- Type of observable partial functions on λ:

$$\sigma^\dagger \triangleq \lambda \to \iota \times \sigma$$

  - $\mathtt{p}_1\,(\mathtt{M}\,\mathtt{N}) \rightsquigarrow^* \mathtt{z}$ iff $\mathtt{M} : \sigma^\dagger$ defined in $\mathtt{N} : \lambda$ with value $\mathtt{p}_2\,(\mathtt{M}\,\mathtt{N})$
- Empty partial function $\{\} : \sigma^\dagger$ s.t. $\{\}\ \mathtt{M} \rightsquigarrow^* \langle \mathtt{s}\,\mathtt{z}, \mathtt{can}_\sigma \rangle$
  - $\mathtt{can}_\sigma : \sigma$ is an inductively defined canonical term

Preliminaries for bar recursion: observable partial functions

- Type of observable partial functions on λ:

$$\sigma^{\dagger} \triangleq \lambda \to \iota \times \sigma$$

  - $\mathtt{p}_1\,(\mathtt{M}\,\mathtt{N}) \rightsquigarrow^* \mathtt{z}$ iff $\mathtt{M} : \sigma^{\dagger}$ defined in $\mathtt{N} : \lambda$ with value $\mathtt{p}_2\,(\mathtt{M}\,\mathtt{N})$
- Empty partial function $\{\} : \sigma^{\dagger}$ s.t. $\{\}\,\mathtt{M} \rightsquigarrow^* \langle \mathtt{s}\,\mathtt{z}, \mathtt{can}_{\sigma} \rangle$
  - $\mathtt{can}_{\sigma} : \sigma$ is an inductively defined canonical term
- $\mathtt{M} \mid \mathtt{N}$ completes $\mathtt{M} : \sigma^{\dagger}$ with $\mathtt{N} : \lambda \to \sigma$, i.e.:

$$\mathtt{M} \mid \mathtt{N} \rightsquigarrow^* \begin{cases} \mathtt{p}_2\,(\mathtt{M}\,\mathtt{P}) & \text{if } \mathtt{p}_1\,(\mathtt{M}\,\mathtt{P}) \rightsquigarrow^* \mathtt{z} \\ \mathtt{N}\,\mathtt{P} & \text{otherwise} \end{cases}$$

Preliminaries for bar recursion: observable partial functions

- Type of observable partial functions on λ:

$$\sigma^\dagger \triangleq \lambda \to \iota \times \sigma$$

  - $\mathtt{p_1}\,(\mathtt{M}\,\mathtt{N}) \rightsquigarrow^* \mathtt{z}$ iff $\mathtt{M} : \sigma^\dagger$ defined in $\mathtt{N} : \lambda$ with value $\mathtt{p_2}\,(\mathtt{M}\,\mathtt{N})$
- Empty partial function $\{\} : \sigma^\dagger$ s.t. $\{\}\ \mathtt{M} \rightsquigarrow^* \langle \mathtt{s}\,\mathtt{z}, \mathtt{can}_\sigma \rangle$
  - $\mathtt{can}_\sigma : \sigma$ is an inductively defined canonical term
- $\mathtt{M} \mid \mathtt{N}$ completes $\mathtt{M} : \sigma^\dagger$ with $\mathtt{N} : \lambda \to \sigma$, i.e.:

$$\mathtt{M} \mid \mathtt{N} \rightsquigarrow^* \begin{cases} \mathtt{p_2}\,(\mathtt{M}\,\mathtt{P}) & \text{if } \mathtt{p_1}\,(\mathtt{M}\,\mathtt{P}) \rightsquigarrow^* \mathtt{z} \\ \mathtt{N}\,\mathtt{P} & \text{otherwise} \end{cases}$$

- $\mathtt{M} \cup \{\mathtt{N} \mapsto \mathtt{P}\}$ extends $\mathtt{M} : \sigma^\dagger$ with $\mathtt{P} : \sigma$ at $\mathtt{N} : \lambda$, i.e.:

$$(\mathtt{M} \cup \{\mathtt{N} \mapsto \mathtt{P}\})\,\mathtt{Q} \rightsquigarrow^* \begin{cases} \langle \mathtt{z}, \mathtt{P} \rangle & \text{if } \mathtt{eq}\,\mathtt{N}\,\mathtt{Q} \rightsquigarrow^* \mathtt{z} \\ \mathtt{M}\,\mathtt{Q} & \text{otherwise} \end{cases}$$

System ΛT_{br}

New constant:

$$\mathtt{brec} : ((\sigma \to \iota) \to \sigma) \to ((\lambda \to \sigma) \to \iota) \to \sigma^{\dagger} \to \iota$$

System ΛT_{br}

New constant:

$$\texttt{brec} : ((\sigma \to \iota) \to \sigma) \to ((\lambda \to \sigma) \to \iota) \to \sigma^{\dagger} \to \iota$$

Reduction:

$$\texttt{brec}\,\mathtt{M}\,\mathtt{N}\,\mathtt{P} \rightsquigarrow \mathtt{N}\,(\mathtt{P} \mid \lambda x.\mathtt{M}\,(\lambda y.\texttt{brec}\,\mathtt{M}\,\mathtt{N}\,(\mathtt{P} \cup \{x \mapsto y\})))$$

System ΛT_{br}

New constant:

$$\texttt{brec} : ((\sigma \to \iota) \to \sigma) \to ((\lambda \to \sigma) \to \iota) \to \sigma^{\dagger} \to \iota$$

Reduction:

$$\texttt{brec}\,\texttt{M}\,\texttt{N}\,\texttt{P} \rightsquigarrow \texttt{N}\,(\texttt{P} \mid \lambda x.\texttt{M}\,(\lambda y.\texttt{brec}\,\texttt{M}\,\texttt{N}\,(\texttt{P} \cup \{x \mapsto y\})))$$

$\texttt{N}$ continuous $\Rightarrow$ looks at only finitely many values of:

$$\texttt{P} \mid \lambda x.\texttt{M}\,(\lambda y.\texttt{brec}\,\texttt{M}\,\texttt{N}\,(\texttt{P} \cup \{x \mapsto y\}))$$

System ΛT_{br}

New constant:

$$\texttt{brec} : ((\sigma \to \iota) \to \sigma) \to ((\lambda \to \sigma) \to \iota) \to \sigma^{\dagger} \to \iota$$

Reduction:

$$\texttt{brec}\,\texttt{M}\,\texttt{N}\,\texttt{P} \rightsquigarrow \texttt{N}\,(\texttt{P} \mid \lambda x.\texttt{M}\,(\lambda y.\texttt{brec}\,\texttt{M}\,\texttt{N}\,(\texttt{P} \cup \{x \mapsto y\})))$$

$\texttt{N}$ continuous $\Rightarrow$ looks at only finitely many values of:

$$\texttt{P} \mid \lambda x.\texttt{M}\,(\lambda y.\texttt{brec}\,\texttt{M}\,\texttt{N}\,(\texttt{P} \cup \{x \mapsto y\}))$$

- if $\texttt{P}$ defined at all these values: same result as $\texttt{N}\,(\texttt{P} \mid \texttt{can}_{\lambda \to \sigma})$
- if $\texttt{N}$ needs value at $\texttt{Q} : \lambda$ and $\texttt{p}_1\,(\texttt{P}\,\texttt{Q}) \not\rightsquigarrow^{*} \texttt{z}$, then call recursively $\texttt{brec}\,\texttt{M}\,\texttt{N}\,(\texttt{P} \cup \{\texttt{Q} \mapsto y\})$ where y is provided by $\texttt{M}$
- It terminates because $\texttt{N}$ is continuous

Domain semantics of system ΛT_{br}

- For each type σ define domain $[\![\sigma]\!]$:

$$[\![\iota]\!] \triangleq \mathbb{N}_\perp \qquad [\![\lambda]\!] \triangleq \Lambda_\perp \qquad [\![\lambda^\diamond]\!] \triangleq (\Lambda^*)_\perp \qquad [\![\top]\!] \triangleq \{*\}_\perp$$

$$[\![\sigma \to \tau]\!] \triangleq \{\varphi : [\![\sigma]\!] \to [\![\tau]\!] \mid \varphi \text{ continuous}\} \qquad [\![\sigma \times \tau]\!] \triangleq [\![\sigma]\!] \times [\![\tau]\!]$$

where:

- $E_\perp$ is $E \cup \{\perp\}$ with $\varphi \leq \psi$ iff $\varphi = \perp$ or $\varphi = \psi$
- $[\![\sigma \to \tau]\!]$ is ordered pointwise
- $[\![\sigma \times \tau]\!]$ is ordered componentwise

Domain semantics of system ΛT_{br}

- For each type σ define domain $[\![\sigma]\!]$:

$$[\![\iota]\!] \triangleq \mathbb{N}_\perp \qquad [\![\lambda]\!] \triangleq \Lambda_\perp \qquad [\![\lambda^\diamond]\!] \triangleq (\Lambda^*)_\perp \qquad [\![\top]\!] \triangleq \{*\}_\perp$$

$$[\![\sigma \to \tau]\!] \triangleq \{\varphi : [\![\sigma]\!] \to [\![\tau]\!] \mid \varphi \text{ continuous}\} \qquad [\![\sigma \times \tau]\!] \triangleq [\![\sigma]\!] \times [\![\tau]\!]$$

  where:
  - $E_\perp$ is $E \cup \{\perp\}$ with $\varphi \leq \psi$ iff $\varphi = \perp$ or $\varphi = \psi$
  - $[\![\sigma \to \tau]\!]$ is ordered pointwise
  - $[\![\sigma \times \tau]\!]$ is ordered componentwise
- For each term $\mathtt{M} : \sigma$ define $[\![\mathtt{M}]\!] \in [\![\sigma]\!]$

Domain semantics of system ΛT_{br}

- For each type σ define domain $[\![\sigma]\!]$:

$$[\![\iota]\!] \triangleq \mathbb{N}_\perp \qquad [\![\lambda]\!] \triangleq \Lambda_\perp \qquad [\![\lambda^\diamond]\!] \triangleq (\Lambda^*)_\perp \qquad [\![\top]\!] \triangleq \{*\}_\perp$$

$$[\![\sigma \to \tau]\!] \triangleq \{\varphi : [\![\sigma]\!] \to [\![\tau]\!] \mid \varphi \text{ continuous}\} \qquad [\![\sigma \times \tau]\!] \triangleq [\![\sigma]\!] \times [\![\tau]\!]$$

  where:
  - $E_\perp$ is $E \cup \{\perp\}$ with $\varphi \leq \psi$ iff $\varphi = \perp$ or $\varphi = \psi$
  - $[\![\sigma \to \tau]\!]$ is ordered pointwise
  - $[\![\sigma \times \tau]\!]$ is ordered componentwise
- For each term $\mathtt{M} : \sigma$ define $[\![\mathtt{M}]\!] \in [\![\sigma]\!]$

We have soundness:

$$\mathtt{M} \rightsquigarrow \mathtt{N} \Rightarrow [\![\mathtt{M}]\!] = [\![\mathtt{N}]\!]$$

Domain semantics of system ΛT_{br}

- For each type σ define domain $[\![\sigma]\!]$:

$$[\![\iota]\!] \triangleq \mathbb{N}_\perp \qquad [\![\lambda]\!] \triangleq \Lambda_\perp \qquad [\![\lambda^\diamond]\!] \triangleq (\Lambda^*)_\perp \qquad [\![\top]\!] \triangleq \{*\}_\perp$$

$$[\![\sigma \to \tau]\!] \triangleq \{\varphi : [\![\sigma]\!] \to [\![\tau]\!] \mid \varphi \text{ continuous}\} \qquad [\![\sigma \times \tau]\!] \triangleq [\![\sigma]\!] \times [\![\tau]\!]$$

  where:
  - $E_\perp$ is $E \cup \{\perp\}$ with $\varphi \leq \psi$ iff $\varphi = \perp$ or $\varphi = \psi$
  - $[\![\sigma \to \tau]\!]$ is ordered pointwise
  - $[\![\sigma \times \tau]\!]$ is ordered componentwise
- For each term $\mathtt{M} : \sigma$ define $[\![\mathtt{M}]\!] \in [\![\sigma]\!]$

We have soundness:

$$\mathtt{M} \rightsquigarrow \mathtt{N} \Rightarrow [\![\mathtt{M}]\!] = [\![\mathtt{N}]\!]$$

and computational adequacy:

$$\mathtt{M} : \iota \wedge [\![\mathtt{M}]\!] = \mathfrak{n} \Rightarrow \mathtt{M} \rightsquigarrow^* \mathtt{s}^n\, \mathtt{z}$$

and similarly on λ and $\lambda^\diamond$

A realizability model for our logic

Mapping logic to system ΛT_{br}

We map terms m, M, Π to programs $m^* : \iota$, $M^* : \lambda$, $\Pi^* : \lambda^\diamond$

- variables i, t, π are variables of system ΛT_{br} of type ι, λ, $\lambda^\diamond$
- $\_^*$ is such that $\mathsf{FV}(\_^*) = \mathsf{FV}(\_)$
- $$i^* = i \qquad 0^* = \mathtt{z} \qquad (S\, m)^* = \mathtt{s}\, m^*$$
$$t^* = t \qquad \underline{m}^* = \mathtt{var}\, m^* \qquad (\lambda.M)^* = \mathtt{abs}\, M^*$$
$$(M\, \Pi)^* = \mathtt{app}^\diamond\, M^*\, \Pi^* \qquad (M\, [m \mapsto \Pi])^* = M^*\, [m^* \mapsto \Pi^*]$$
$$\pi^* = \pi \qquad \langle\rangle^* = \mathtt{nil} \qquad \langle \Pi, M\rangle^* = \mathtt{cons}\, \Pi^*\, M^*$$
- No X^*, b^* because no $(\!|X|\!)$, $(\!|b|\!)$: X, Φ are not computational

Mapping logic to system ΛT_{br}

We map terms m, M, Π to programs $m^* : \iota$, $M^* : \lambda$, $\Pi^* : \lambda^\diamond$

- variables i, t, π are variables of system ΛT_{br} of type ι, λ, $\lambda^\diamond$
- $\_^*$ is such that $\mathrm{FV}(\_^*) = \mathrm{FV}(\_)$
- $$i^* = i \qquad 0^* = \mathtt{z} \qquad (S\, m)^* = \mathtt{s}\, m^*$$
  $$t^* = t \qquad \underline{m}^* = \mathtt{var}\, m^* \qquad (\lambda.M)^* = \mathtt{abs}\, M^*$$
  $$(M\,\Pi)^* = \mathtt{app}^\diamond\, M^*\, \Pi^* \qquad (M\,[m \mapsto \Pi])^* = M^*\,[m^* \mapsto \Pi^*]$$
  $$\pi^* = \pi \qquad \langle\rangle^* = \mathtt{nil} \qquad \langle \Pi, M\rangle^* = \mathtt{cons}\, \Pi^*\, M^*$$
- No X^*, b^* because no $(\!|X|\!)$, $(\!|b|\!)$: X, Φ are not computational

We map formulas A to types A^* of system ΛT_{br}

- $$\Phi^* = \iota \qquad (M{\downarrow}^m)^* = \top \qquad (\forall_\_ A)^* = A^*$$
  $$(\!|m|\!)^* = \iota \qquad (\!|M|\!)^* = \lambda \qquad (\!|\Pi|\!)^* = \lambda^\diamond$$
  $$(A \Rightarrow B)^* = A^* \to B^* \qquad (A \wedge B)^* = A^* \times B^*$$
- $(M{\downarrow}^m)^* = \top$: $M{\downarrow}^m$ is computationally irrelevant
- $\Phi^* = \iota$: we extract nat. numbers (bounds on reduction steps)
- $\forall$ erased: quantifications are uniform by default

Formulas with parameters

- Closed formulas/terms with parameters: formulas/terms where free variables are replaced by real-world elements:

$$i \text{ are replaced with } \mathfrak{n} \in \mathbb{N} \qquad t \text{ with } \mathfrak{M} \in \Lambda \qquad \pi \text{ with } \mathbf{\Pi} \in \Lambda^*$$

$$X \text{ with } \mathfrak{X} \in \mathcal{P}(\Lambda) \qquad b \text{ with } \mathfrak{b} \in \{\mathfrak{tt}; \mathfrak{ff}\}$$

Formulas with parameters

- Closed formulas/terms with parameters: formulas/terms where free variables are replaced by real-world elements:

  i are replaced with $\mathfrak{n} \in \mathbb{N}$ $\quad$ t with $\mathfrak{M} \in \Lambda$ $\quad$ π with $\mathbf{\Pi} \in \Lambda^*$

  X with $\mathfrak{X} \in \mathcal{P}(\Lambda)$ $\quad$ b with $\mathfrak{b} \in \{\mathfrak{tt}; \mathfrak{ff}\}$

- Since $\mathbb{N} \subseteq \mathbb{N}_\perp$, $\Lambda \subseteq \Lambda_\perp$ and $\Lambda^* \subseteq (\Lambda^*)_\perp$:

  if m, M, Π closed terms with parameters

  then $[\![m^*]\!] \in [\![\iota]\!], [\![M^*]\!] \in [\![\lambda]\!], [\![\Pi^*]\!] \in [\![\lambda^\diamond]\!]$

Formulas with parameters

- Closed formulas/terms with parameters: formulas/terms where free variables are replaced by real-world elements:

$$i \text{ are replaced with } \mathfrak{n} \in \mathbb{N} \qquad t \text{ with } \mathfrak{M} \in \Lambda \qquad \pi \text{ with } \mathbf{\Pi} \in \Lambda^*$$
$$X \text{ with } \mathfrak{X} \in \mathcal{P}(\Lambda) \qquad b \text{ with } \mathfrak{b} \in \{\mathfrak{tt}; \mathfrak{ff}\}$$

- Since $\mathbb{N} \subseteq \mathbb{N}_\bot$, $\Lambda \subseteq \Lambda_\bot$ and $\Lambda^* \subseteq (\Lambda^*)_\bot$:

if m, M, Π closed terms with parameters

$$\text{then } [\![m^*]\!] \in [\![\iota]\!], [\![M^*]\!] \in [\![\lambda]\!], [\![\Pi^*]\!] \in [\![\lambda^\diamond]\!]$$

- Closed formula A with parameters gets a realizability value:

$$|A| \subseteq [\![A^*]\!]$$

Formulas with parameters

- Closed formulas/terms with parameters: formulas/terms where free variables are replaced by real-world elements:

$$i \text{ are replaced with } \mathfrak{n} \in \mathbb{N} \qquad t \text{ with } \mathfrak{M} \in \Lambda \qquad \pi \text{ with } \mathbf{\Pi} \in \Lambda^*$$
$$X \text{ with } \mathfrak{X} \in \mathcal{P}(\Lambda) \qquad b \text{ with } \mathfrak{b} \in \{\mathfrak{tt}; \mathfrak{ff}\}$$

- Since $\mathbb{N} \subseteq \mathbb{N}_\perp$, $\Lambda \subseteq \Lambda_\perp$ and $\Lambda^* \subseteq (\Lambda^*)_\perp$:

  if m, M, Π closed terms with parameters

  then $[\![m^*]\!] \in [\![\iota]\!], [\![M^*]\!] \in [\![\lambda]\!], [\![\Pi^*]\!] \in [\![\lambda^\diamond]\!]$

- Closed formula A with parameters gets a realizability value:

$$|A| \subseteq [\![A^*]\!]$$

- The model is parameterized by a pole:

$$\perp\!\!\!\perp \subseteq \mathbb{N}$$

  we extract natural numbers (bounds on reduction steps)

Realizability values: atomic predicates

- $$|\mathfrak{tt}| = |tt| = \mathbb{N}_\perp \qquad |M \in \mathfrak{X}| = \begin{cases} \mathbb{N}_\perp & \text{if } [\![M^*]\!] \in \mathfrak{X} \\ \perp\!\!\!\perp & \text{if } [\![M^*]\!] \notin \mathfrak{X} \end{cases}$$
  $$|\mathfrak{ff}| = |ff| = \perp\!\!\!\perp$$
  - these atomic predicates are computationally relevant $\rightsquigarrow$ their realizability values depend on $\perp\!\!\!\perp$

Realizability values: atomic predicates

- $$|\mathfrak{tt}| = |tt| = \mathbb{N}_\perp \qquad |M \in \mathfrak{X}| = \begin{cases} \mathbb{N}_\perp & \text{if } [\![M^*]\!] \in \mathfrak{X} \\ \perp\!\!\!\perp & \text{if } [\![M^*]\!] \notin \mathfrak{X} \end{cases}$$
  $$|\mathfrak{ff}| = |f\!f| = \perp\!\!\!\perp$$
  - these atomic predicates are computationally relevant
    $\rightsquigarrow$ their realizability values depend on $\perp\!\!\!\perp$
- $$|M{\downarrow}^m| = \begin{cases} \{*\}_\perp & \text{if } [\![M^*]\!] \text{ normalizes in at most } [\![m^*]\!] \text{ steps} \\ \emptyset & \text{otherwise} \end{cases}$$
  - these are computationally irrelevant
    $\rightsquigarrow$ their realizability values are independent from $\perp\!\!\!\perp$

Realizability values: atomic predicates

- $|\mathfrak{tt}| = |tt| = \mathbb{N}_\perp$ $\quad |\mathfrak{ff}| = |ff| = \perp\!\!\!\perp$ $\qquad |M \in \mathfrak{X}| = \begin{cases} \mathbb{N}_\perp & \text{if } [\![M^*]\!] \in \mathfrak{X} \\ \perp\!\!\!\perp & \text{if } [\![M^*]\!] \notin \mathfrak{X} \end{cases}$
  - these atomic predicates are computationally relevant
    $\rightsquigarrow$ their realizability values depend on $\perp\!\!\!\perp$
- $|M{\downarrow}^m| = \begin{cases} \{*\}_\perp & \text{if } [\![M^*]\!] \text{ normalizes in at most } [\![m^*]\!] \text{ steps} \\ \emptyset & \text{otherwise} \end{cases}$
  - these are computationally irrelevant
    $\rightsquigarrow$ their realizability values are independent from $\perp\!\!\!\perp$
- $|(\!|m|\!)| = \{[\![m^*]\!]\}$ $\qquad |(\!|M|\!)| = \{[\![M^*]\!]\}$ $\qquad |(\!|\Pi|\!)| = \{[\![\Pi^*]\!]\}$
  - these are relativizations
    $\rightsquigarrow$ only one realizer: the value of the enclosed term

Realizability values: connectives

- $|A \Rightarrow B| = \{\varphi \in [\![A^* \to B^*]\!] \mid \forall \psi \in |A|, \varphi(\psi) \in |B|\}$

  $|A \wedge B| = \{(\varphi, \psi) \in [\![A^* \times B^*]\!] \mid \varphi \in |A| \wedge \psi \in |B|\}$

  - standard definitions

Realizability values: connectives

- $|A \Rightarrow B| = \{\varphi \in [\![A^* \to B^*]\!] \mid \forall \psi \in |A|, \varphi(\psi) \in |B|\}$

  $|A \wedge B| = \{(\varphi, \psi) \in [\![A^* \times B^*]\!] \mid \varphi \in |A| \wedge \psi \in |B|\}$
  - standard definitions

- $$|\forall i\, A| = \bigcap_{\mathfrak{n} \in \mathbb{N}} |A[\mathfrak{n}/i]| \quad |\forall t\, A| = \bigcap_{\mathfrak{M} \in \Lambda} |A[\mathfrak{M}/t]|$$

  $$|\forall \pi\, A| = \bigcap_{\mathbf{\Pi} \in \Lambda^*} |A[\mathbf{\Pi}/\pi]|$$

  $$|\forall X\, A| = \bigcap_{\mathfrak{X} \in \mathcal{P}(\Lambda)} |A[\mathfrak{X}/X]| \qquad |\forall b\, A| = \bigcap_{\mathfrak{b} \in \{\mathfrak{tt}; \mathfrak{ff}\}} |A[\mathfrak{b}/b]|$$
  - quantified formulas are instantiated with real-world elements

A restricted double-negation elimination

double-negation elimination on A is $\neg\neg A \Rightarrow A$

A restricted double-negation elimination

double-negation elimination on A is $\neg\neg A \Rightarrow A$

- unrealizable for computationally irrelevant formula $M{\downarrow}^m$:
  - if $\perp\!\!\!\perp \neq \emptyset$
  - and if $[\![M^*]\!]$ doesn't normalize after $[\![m^*]\!]$ steps
  - then $|\neg\neg M{\downarrow}^m| \neq \emptyset$ and $|M{\downarrow}^m| = \emptyset$, so $|\neg\neg M{\downarrow}^m \Rightarrow M{\downarrow}^m| = \emptyset$

A restricted double-negation elimination

double-negation elimination on A is $\neg\neg A \Rightarrow A$

- unrealizable for computationally irrelevant formula $M{\downarrow}^m$:
  - if $⫫ \neq \emptyset$
  - and if $[\![M^*]\!]$ doesn't normalize after $[\![m^*]\!]$ steps
  - then $|\neg\neg M{\downarrow}^m| \neq \emptyset$ and $|M{\downarrow}^m| = \emptyset$, so $|\neg\neg M{\downarrow}^m \Rightarrow M{\downarrow}^m| = \emptyset$
- but realizable for negative formulas:

$$A^-, B^- \quad ::= \quad \Phi \mid A \Rightarrow B^- \mid A^- \wedge B^- \mid \forall_{\_} A^-$$

A restricted double-negation elimination

double-negation elimination on A is $\neg\neg A \Rightarrow A$

- unrealizable for computationally irrelevant formula $M{\downarrow}^m$:
  - if $\perp\!\!\!\perp \neq \emptyset$
  - and if $[\![M^*]\!]$ doesn't normalize after $[\![m^*]\!]$ steps
  - then $|\neg\neg M{\downarrow}^m| \neq \emptyset$ and $|M{\downarrow}^m| = \emptyset$, so $|\neg\neg M{\downarrow}^m \Rightarrow M{\downarrow}^m| = \emptyset$
- but realizable for negative formulas:

$$A^-, B^- \quad ::= \quad \Phi \mid A \Rightarrow B^- \mid A^- \wedge B^- \mid \forall_\_ A^-$$

- $[\![\mathtt{dne}_{A^-}]\!] \in |\neg\neg A^- \Rightarrow A^-|$ where:

$$\mathtt{dne}_\Phi = \lambda x.x\,(\lambda y.y) \qquad \mathtt{dne}_{\forall_\epsilon A^-} = \mathtt{dne}_{A^-}$$

$$\mathtt{dne}_{A \Rightarrow B^-} = \lambda xy.\mathtt{dne}_{B^-}\,(\lambda z.x\,(\lambda u.z\,(u\,y)))$$

$$\mathtt{dne}_{A^- \wedge B^-} = \lambda x.\,\langle \mathtt{dne}_{A^-}\,(\lambda y.x\,(\lambda z.y\,(\mathrm{p}_1\,z)))\,, \mathtt{dne}_{B^-}\,(\lambda y.x\,(\lambda z.y\,(\mathrm{p}_2\,z)))\rangle$$

A restricted double-negation elimination

double-negation elimination on A is $\neg\neg A \Rightarrow A$

- unrealizable for computationally irrelevant formula $M{\downarrow}^m$:
  - if $\perp\!\!\!\perp \neq \emptyset$
  - and if $[\![M^*]\!]$ doesn't normalize after $[\![m^*]\!]$ steps
  - then $|\neg\neg M{\downarrow}^m| \neq \emptyset$ and $|M{\downarrow}^m| = \emptyset$, so $|\neg\neg M{\downarrow}^m \Rightarrow M{\downarrow}^m| = \emptyset$
- but realizable for negative formulas:

$$A^-, B^- \quad ::= \quad \Phi \mid A \Rightarrow B^- \mid A^- \wedge B^- \mid \forall_{\_} A^-$$

- $[\![\mathtt{dne}_{A^-}]\!] \in |\neg\neg A^- \Rightarrow A^-|$ where:

$$\mathtt{dne}_\Phi = \lambda x.x\,(\lambda y.y) \qquad \mathtt{dne}_{\forall_\epsilon A^-} = \mathtt{dne}_{A^-}$$

$$\mathtt{dne}_{A \Rightarrow B^-} = \lambda xy.\mathtt{dne}_{B^-}\,(\lambda z.x\,(\lambda u.z\,(u\,y)))$$

$$\mathtt{dne}_{A^- \wedge B^-} = \lambda x.\,\langle \mathtt{dne}_{A^-}\,(\lambda y.x\,(\lambda z.y\,(\mathrm{p}_1\,z)))\,, \mathtt{dne}_{B^-}\,(\lambda y.x\,(\lambda z.y\,(\mathrm{p}_2\,z)))\rangle$$

- $[\![\mathtt{dne}_\Phi]\!] \in |\neg\neg\Phi \Rightarrow \Phi|$ by disjunction of cases

Interpreting second-order elimination

- In Krivine's realizability, $\lambda x.x \Vdash \forall X\, A(X) \Rightarrow A(B)$
  - proof terms are polymorphic

Interpreting second-order elimination

- In Krivine's realizability, $\lambda x.x \Vdash \forall X\, A(X) \Rightarrow A(B)$
  - proof terms are polymorphic
- here, proof terms (and realizers) are simply-typed

Interpreting second-order elimination

- In Krivine's realizability, $\lambda x.x \Vdash \forall X\, A(X) \Rightarrow A(B)$
  - proof terms are polymorphic
- here, proof terms (and realizers) are simply-typed
- $\forall X\, A(X) \Rightarrow A(B)$ interpreted in two steps:

Interpreting second-order elimination

- In Krivine's realizability, $\lambda x.x \Vdash \forall X\, A(X) \Rightarrow A(B)$
  - proof terms are polymorphic
- here, proof terms (and realizers) are simply-typed
- $\forall X\, A(X) \Rightarrow A(B)$ interpreted in two steps:
  - comprehension: $\exists X\, \forall^r t\, (t \in X \Leftrightarrow B^-(t))$
    - B^- negative
    - relativized quantification on t
    - using bar recursion

Interpreting second-order elimination

- In Krivine's realizability, $\lambda x.x \Vdash \forall X\, A(X) \Rightarrow A(B)$
  - proof terms are polymorphic
- here, proof terms (and realizers) are simply-typed
- $\forall X\, A(X) \Rightarrow A(B)$ interpreted in two steps:
  - comprehension: $\exists X\, \forall^r t\, (t \in X \Leftrightarrow B^- (t))$
    - B^- negative
    - relativized quantification on t
    - using bar recursion
  - $\forall^r t\, (B(t) \Leftrightarrow C(t)) \Rightarrow (A^r(B) \Leftrightarrow A^r(C))$
    - A^r relativized:

$$\begin{aligned} A^r, B^r \quad ::= \quad & P \mid A^r \Rightarrow B^r \mid A^r \wedge B^r \\ & \mid \forall^r i\, A^r \mid \forall^r t\, A^r \mid \forall^r \pi\, A^r \mid \forall X\, A^r \mid \forall b\, A^r \mid \end{aligned}$$

    - realizer defined by induction on A

Interpreting second-order elimination

- In Krivine's realizability, $\lambda x.x \Vdash \forall X\, A(X) \Rightarrow A(B)$
  - proof terms are polymorphic
- here, proof terms (and realizers) are simply-typed
- $\forall X\, A(X) \Rightarrow A(B)$ interpreted in two steps:
  - comprehension: $\exists X\, \forall^r t\, (t \in X \Leftrightarrow B^-(t))$
    - B^- negative
    - relativized quantification on t
    - using bar recursion
  - $\forall^r t\, (B(t) \Leftrightarrow C(t)) \Rightarrow (A^r(B) \Leftrightarrow A^r(C))$
    - A^r relativized:

$$\begin{aligned} A^r, B^r \quad ::= \quad & P \mid A^r \Rightarrow B^r \mid A^r \wedge B^r \\ & \mid \forall^r i\, A^r \mid \forall^r t\, A^r \mid \forall^r \pi\, A^r \mid \forall X\, A^r \mid \forall b\, A^r \mid \end{aligned}$$

    - realizer defined by induction on A
- finally, we will get $\forall X\, A^r\left(\overline{X}\right) \Rightarrow A^r\left(B^-\right)^-$

Comprehension: $\exists X \, \forall^r t \, (t \in X \Leftrightarrow B^- (t))$

Two steps:

Comprehension: $\exists X \, \forall^r t \, (t \in X \Leftrightarrow B^- (t))$

Two steps:

- $[\![\lambda xy.\mathtt{brec}\,(\lambda z.\mathtt{exf}_{A^-}\,(x\,z))\,y\,\{\}]\!]$
  $$\in \left|\forall t \, \exists b \, A^- (b, t) \Rightarrow \exists X \, \forall^r t \, A^- (t \in X, t)\right|$$
  - quite technical, uses Zorn's lemma

Comprehension: $\exists X\,\forall^r t\,(t \in X \Leftrightarrow B^-(t))$

Two steps:

- $[\![\lambda xy.\mathtt{brec}\,(\lambda z.\mathtt{exf}_{A^-}\,(x\,z))\,y\,\{\}]\!]$
  $\in \left|\forall t\,\exists b\,A^-(b,t) \Rightarrow \exists X\,\forall^r t\,A^-(t \in X, t)\right|$
  - quite technical, uses Zorn's lemma
- $[\![\lambda x.x\,\langle \mathtt{exf}_{A^-}, \lambda y.x\,\langle \lambda\_.y, \lambda\_.\mathtt{z}\rangle\rangle]\!] \in |\forall t\,\exists b\,(b \Leftrightarrow A^-(t))|$
  - quite straightforward:

Comprehension: $\exists X\, \forall^r t\, (t \in X \Leftrightarrow B^-(t))$

Two steps:

- $[\![\lambda xy.\texttt{brec}\,(\lambda z.\texttt{exf}_{A^-}\,(x\,z))\,y\,\{\}]\!]$

  $$\in \left|\forall t\, \exists b\, A^-(b, t) \Rightarrow \exists X\, \forall^r t\, A^-(t \in X, t)\right|$$

  - quite technical, uses Zorn's lemma
- $[\![\lambda x.x\,\langle \texttt{exf}_{A^-}, \lambda y.x\,\langle \lambda\_.y, \lambda\_.\mathtt{z}\rangle\rangle]\!] \in |\forall t\, \exists b\, (b \Leftrightarrow A^-(t))|$
  - quite straightforward:
    Let $\mathfrak{M} \in \Lambda$ and $\varphi \in |\forall b\, \neg (b \Leftrightarrow A^-(\mathfrak{M}))|$. We prove:

    $$[\![\varphi\,\langle \texttt{exf}_{A^-}, \lambda y.\varphi\,\langle \lambda\_.y, \lambda\_.\mathtt{z}\rangle\rangle]\!] \in |\mathit{ff}|$$

Comprehension: $\exists X\, \forall^r t\, (t \in X \Leftrightarrow B^-(t))$

Two steps:

- $[\![\lambda xy.\mathtt{brec}\,(\lambda z.\mathtt{exf}_{A^-}\,(x\,z))\,y\,\{\}]\!]$
$$\in \left|\forall t\, \exists b\, A^-(b,t) \Rightarrow \exists X\, \forall^r t\, A^-(t \in X, t)\right|$$
  - quite technical, uses Zorn's lemma
- $[\![\lambda x.x\,\langle \mathtt{exf}_{A^-}, \lambda y.x\,\langle \lambda\_.y, \lambda\_.\mathrm{z}\rangle\rangle]\!] \in |\forall t\, \exists b\, (b \Leftrightarrow A^-(t))|$
  - quite straightforward:
    Let $\mathfrak{M} \in \Lambda$ and $\varphi \in |\forall b\, \neg\, (b \Leftrightarrow A^-(\mathfrak{M}))|$. We prove:
$$[\![\varphi\,\langle \mathtt{exf}_{A^-}, \lambda y.\varphi\,\langle \lambda\_.y, \lambda\_.\mathrm{z}\rangle\rangle]\!] \in |\mathit{ff}|$$
    $\varphi \in |\neg\, (\mathit{ff} \Leftrightarrow A^-(\mathfrak{M}))|$ so we need to prove:
    $[\![\mathtt{exf}_{A^-}]\!] \in \left|\mathit{ff} \Rightarrow A^-(\mathfrak{M})\right|$ and $[\![\lambda y.\varphi\,\langle \lambda\_.y, \lambda\_.\mathrm{z}\rangle]\!] \in \left|A^-(\mathfrak{M}) \Rightarrow \mathit{ff}\right|$

Comprehension: $\exists X\,\forall^r t\,(t \in X \Leftrightarrow B^-(t))$

Two steps:

- $[\![\lambda xy.\mathtt{brec}\,(\lambda z.\mathtt{exf}_{A^-}\,(x\,z))\,y\,\{\}]\!]$
  $$\in \left|\forall t\,\exists b\,A^-(b,t) \Rightarrow \exists X\,\forall^r t\,A^-(t \in X, t)\right|$$
  - quite technical, uses Zorn's lemma
- $[\![\lambda x.x\,\langle \mathtt{exf}_{A^-}, \lambda y.x\,\langle \lambda\_.y, \lambda\_.\mathtt{z}\rangle\rangle]\!] \in |\forall t\,\exists b\,(b \Leftrightarrow A^-(t))|$
  - quite straightforward:
    Let $\mathfrak{M} \in \Lambda$ and $\varphi \in |\forall b\,\neg\,(b \Leftrightarrow A^-(\mathfrak{M}))|$. We prove:
    $$[\![\varphi\,\langle \mathtt{exf}_{A^-}, \lambda y.\varphi\,\langle \lambda\_.y, \lambda\_.\mathtt{z}\rangle\rangle]\!] \in |f\!f|$$
    $\varphi \in |\neg\,(f\!f \Leftrightarrow A^-(\mathfrak{M}))|$ so we need to prove:

    $[\![\mathtt{exf}_{A^-}]\!] \in \left|f\!f \Rightarrow A^-(\mathfrak{M})\right|$ and $[\![\lambda y.\varphi\,\langle \lambda\_.y, \lambda\_.\mathtt{z}\rangle]\!] \in \left|A^-(\mathfrak{M}) \Rightarrow f\!f\right|$

    Let $\psi \in |A^-(\mathfrak{M})|$, $\varphi \in |\neg\,(tt \Leftrightarrow A^-(\mathfrak{M}))|$ so we prove:
    $$[\![\lambda\_.\psi]\!] \in \left|tt \Rightarrow A^-(\mathfrak{M})\right| \text{ and } [\![\lambda\_.\mathtt{z}]\!] \in \left|A^-(\mathfrak{M}) \Rightarrow tt\right|$$

A weak form of bar recursion

- Our bar recursion:

$$\texttt{brec} : ((\sigma \to \iota) \to \sigma) \to ((\lambda \to \sigma) \to \iota) \to \sigma^\dagger \to \iota$$

$$\texttt{brec}\,\texttt{M}\,\texttt{N}\,\texttt{P} \rightsquigarrow \texttt{N}\,(\texttt{P} \mid \lambda x.\texttt{M}\,(\lambda y.\texttt{brec}\,\texttt{M}\,\texttt{N}\,(\texttt{P} \cup \{x \mapsto y\})))$$

$$\text{realizes } \forall t\, \exists b\, A^-\,(b, t) \Rightarrow \exists X\, \forall^r t\, A^-\,(t \in X, t)$$

A weak form of bar recursion

- Our bar recursion:

$$\texttt{brec} : ((\sigma \to \iota) \to \sigma) \to ((\lambda \to \sigma) \to \iota) \to \sigma^{\dagger} \to \iota$$

$$\texttt{brec}\,\texttt{M}\,\texttt{N}\,\texttt{P} \rightsquigarrow \texttt{N}\,(\texttt{P} \mid \lambda x.\texttt{M}\,(\lambda y.\texttt{brec}\,\texttt{M}\,\texttt{N}\,(\texttt{P} \cup \{x \mapsto y\})))$$

$$\text{realizes } \forall t\, \exists b\, A^{-}\,(b, t) \Rightarrow \exists X\, \forall^{r} t\, A^{-}\,(t \in X, t)$$

- Usual bar recursion:

$$\texttt{brec}' : (\lambda \to (\sigma \to \iota) \to \sigma) \to ((\lambda \to \sigma) \to \iota) \to \sigma^{\dagger} \to \iota$$

$$\texttt{brec}'\,\texttt{M}\,\texttt{N}\,\texttt{P} \rightsquigarrow \texttt{N}\left(\texttt{P} \mid \lambda x.\texttt{M}\,x\left(\lambda y.\texttt{brec}'\,\texttt{M}\,\texttt{N}\,(\texttt{P} \cup \{x \mapsto y\})\right)\right)$$

$$\text{realizes } \forall^{r} t\, \exists b\, A^{-}\,(b, t) \Rightarrow \exists X\, \forall^{r} t\, A^{-}\,(t \in X, t)$$

stronger

A weak form of bar recursion

- Our bar recursion:

$$\texttt{brec} : ((\sigma \to \iota) \to \sigma) \to ((\lambda \to \sigma) \to \iota) \to \sigma^\dagger \to \iota$$

$$\texttt{brec}\,\texttt{M}\,\texttt{N}\,\texttt{P} \rightsquigarrow \texttt{N}\,(\texttt{P} \mid \lambda x.\texttt{M}\,(\lambda y.\texttt{brec}\,\texttt{M}\,\texttt{N}\,(\texttt{P} \cup \{x \mapsto y\})))$$

realizes $\forall t\, \exists b\, A^-\,(b, t) \Rightarrow \exists X\, \forall^r t\, A^-\,(t \in X, t)$

- Usual bar recursion:

$$\texttt{brec}' : (\lambda \to (\sigma \to \iota) \to \sigma) \to ((\lambda \to \sigma) \to \iota) \to \sigma^\dagger \to \iota$$

$$\texttt{brec}'\,\texttt{M}\,\texttt{N}\,\texttt{P} \rightsquigarrow \texttt{N}\left(\texttt{P} \mid \lambda x.\texttt{M}\,x\left(\lambda y.\texttt{brec}'\,\texttt{M}\,\texttt{N}\,(\texttt{P} \cup \{x \mapsto y\})\right)\right)$$

realizes $\forall^r t\, \exists b\, A^-\,(b, t) \Rightarrow \exists X\, \forall^r t\, A^-\,(t \in X, t)$

stronger

Countable choice stronger than comprehension?

$\forall^r t\,(B\,(t) \Leftrightarrow C\,(t)) \Rightarrow (A^r\,(B) \Leftrightarrow A^r\,(C))$

If $\vec{\mathfrak{n}} \in \vec{\mathbb{N}}$, $\vec{\mathfrak{M}} \in \vec{\Lambda}$, $\vec{\mathbf{\Pi}} \in \vec{\Lambda^*}$ then:

$$\left[\!\left[\mathtt{repl}_{A^r}\left[\vec{\mathfrak{n}}/\vec{i}, \vec{\mathfrak{M}}/\vec{t}, \vec{\mathbf{\Pi}}/\vec{\pi}\right]\right]\!\right]$$
$$\in \left|(\forall^r t\,(B\,(t) \Leftrightarrow C\,(t)) \Rightarrow (A^r\,(B) \Leftrightarrow A^r\,(C)))\left[\vec{\mathfrak{n}}/\vec{i}, \vec{\mathfrak{M}}/\vec{t}, \vec{\mathbf{\Pi}}/\vec{\pi}\right]\right|$$

$\forall^r t\,(B\,(t) \Leftrightarrow C\,(t)) \Rightarrow (A^r\,(B) \Leftrightarrow A^r\,(C))$

If $\vec{\mathfrak{n}} \in \vec{\mathbb{N}}$, $\vec{\mathfrak{M}} \in \vec{\Lambda}$, $\vec{\mathbf{\Pi}} \in \vec{\Lambda^*}$ then:

$$\left[\!\left[\mathtt{repl}_{A^r}\left[\vec{\mathfrak{n}}/\vec{i}, \vec{\mathfrak{M}}/\vec{t}, \vec{\mathbf{\Pi}}/\vec{\pi}\right]\right]\!\right]$$
$$\in \left|(\forall^r t\,(B\,(t) \Leftrightarrow C\,(t)) \Rightarrow (A^r\,(B) \Leftrightarrow A^r\,(C)))\left[\vec{\mathfrak{n}}/\vec{i}, \vec{\mathfrak{M}}/\vec{t}, \vec{\mathbf{\Pi}}/\vec{\pi}\right]\right|$$

where $\mathtt{repl}_{A^r} \triangleq \lambda x.\mathtt{repl}'_{A^r}$ and:

$$\mathtt{repl}'_{\overline{X \mapsto M \in X}} = x\,M^* \qquad \mathtt{repl}'_{\overline{X \mapsto P}} = \langle \lambda y.y, \lambda y.y \rangle \text{ if } P \neq M \in X$$

$$\mathtt{repl}'_{A_1^r \Rightarrow A_2^r} = \left\langle \lambda yz.\mathrm{p}_1\,\mathtt{repl}'_{A_2^r}\left(y\left(\mathrm{p}_2\,\mathtt{repl}'_{A_1^r}\,z\right)\right), \lambda yz.\mathrm{p}_2\,\mathtt{repl}'_{A_2^r}\left(y\left(\mathrm{p}_1\,\mathtt{repl}'_{A_1^r}\,z\right)\right)\right\rangle$$

$$\mathtt{repl}'_{A_1^r \wedge A_2^r} = \left\langle \lambda y.\left\langle \mathrm{p}_1\,\mathtt{repl}'_{A_1^r}\,(\mathrm{p}_1\,y), \mathrm{p}_1\,\mathtt{repl}'_{A_2^r}\,(\mathrm{p}_2\,y)\right\rangle, \lambda y.\left\langle \mathrm{p}_2\,\mathtt{repl}'_{A_1^r}\,(\mathrm{p}_1\,y), \mathrm{p}_2\,\mathtt{repl}'_{A_2^r}\,(\mathrm{p}_2\,y)\right\rangle\right\rangle$$

$$\mathtt{repl}'_{\forall^r \eta\,A^r} = \langle \lambda y\eta.\mathrm{p}_1\,\mathtt{repl}'_{A^r}\,(y\,\eta), \lambda y\eta.\mathrm{p}_2\,\mathtt{repl}'_{A^r}\,(y\,\eta)\rangle \qquad \mathtt{repl}'_{\forall X\,A^r} = \mathtt{repl}'_{\forall b\,A^r} = \mathtt{repl}'_{A^r}$$

Second-order elimination

$$\mathtt{elim}_{A^r,B^-} \triangleq$$

$$\lambda x.\mathtt{dne}_{A^r(B^-)^-}\begin{pmatrix} \lambda y.\mathtt{brec}\ (\lambda z.\mathtt{exf}_{B^-}\,(z\,\langle \mathtt{exf}_{B^-}, \lambda u.z\,\langle \lambda\_.u, \lambda\_.\mathrm{z}\rangle\rangle)) \\ (\lambda z.y\,(\mathrm{p}_1\,(\mathtt{repl}_{A^r}\,z)\,x)) \\ \{\} \end{pmatrix}$$

Second-order elimination

$$\mathtt{elim}_{A^r,B^-} \triangleq$$

$$\lambda x.\mathtt{dne}_{A^r(B^-)^-}\begin{pmatrix} \lambda y.\mathtt{brec}\ (\lambda z.\mathtt{exf}_{B^-}\,(z\,\langle \mathtt{exf}_{B^-}, \lambda u.z\,\langle \lambda\_.u, \lambda\_.\mathtt{z}\rangle\rangle)) \\ (\lambda z.y\,(\mathtt{p}_1\,(\mathtt{repl}_{A^r}\,z)\,x)) \\ \{\} \end{pmatrix}$$

$$\mathtt{elim}_{A^r,B^-} \in \left|\forall X\, A^r\left(\overline{X}\right) \Rightarrow A^r\left(B^-\right)^-\right|$$

Second-order elimination

$$\texttt{elim}_{A^r,B^-} \triangleq$$

$$\lambda x.\texttt{dne}_{A^r(B^-)^-}\begin{pmatrix} \lambda y.\texttt{brec}\ (\lambda z.\texttt{exf}_{B^-}\ (z\ \langle \texttt{exf}_{B^-}, \lambda u.z\ \langle \lambda\_.u, \lambda\_.\mathtt{z}\rangle\rangle)) \\ (\lambda z.y\ (\mathrm{p}_1\ (\texttt{repl}_{A^r}\ z)\ x)) \\ \{\} \end{pmatrix}$$

$$\texttt{elim}_{A^r,B^-} \in \left| \forall X\, A^r\left(\overline{X}\right) \Rightarrow A^r\left(B^-\right)^- \right|$$

Believe me!

The realizability interpretation of normalization of $M : T$

Normalization of system F: reminder

Three steps:

Normalization of system F: reminder

Three steps:

- $\mathcal{R}ed\mathcal{C}and\,(\Downarrow)$ where $\Downarrow(t) \triangleq t{\downarrow}$

Normalization of system F: reminder

Three steps:

- $\mathcal{R}ed\mathcal{C}and\,(\Downarrow)$ where $\Downarrow(t) \triangleq t\downarrow$
- If T type of F with $\mathrm{FV}(T) = \{X_0, \ldots, X_{n-1}\}$ then:

$$\forall X_0(\mathcal{R}ed\mathcal{C}and\left(\overline{X_0}\right) \Rightarrow \ldots \Rightarrow \forall X_{n-1}(\mathcal{R}ed\mathcal{C}and\left(\overline{X_{n-1}}\right) \Rightarrow \mathcal{R}ed\mathcal{C}and\,(RC_T)) \ldots)$$

Normalization of system F: reminder

Three steps:

- $\mathcal{R}ed\mathcal{C}and\,(\Downarrow)$ where $\Downarrow(t) \stackrel{\Delta}{=} t\downarrow$
- If T type of F with $\mathrm{FV}(T) = \{X_0, \ldots, X_{n-1}\}$ then:

$$\forall X_0(\mathcal{R}ed\mathcal{C}and\left(\overline{X_0}\right) \Rightarrow \ldots \Rightarrow \forall X_{n-1}(\mathcal{R}ed\mathcal{C}and\left(\overline{X_{n-1}}\right) \Rightarrow \mathcal{R}ed\mathcal{C}and\,(RC_T))\ldots)$$

- If $\mathrm{FV}(T_0, \ldots, T_{m-1}, U) \subseteq \{X_0, \ldots, X_{n-1}\}$ and $T_{m-1}, \ldots, T_0 \vdash M : U$ typing derivation in F then:

$$\forall X_0(\mathcal{R}ed\mathcal{C}and\left(\overline{X_0}\right) \Rightarrow \ldots \Rightarrow \forall X_{n-1}(\mathcal{R}ed\mathcal{C}and\left(\overline{X_{n-1}}\right) \Rightarrow \forall^r t_{m-1}(RC_{T_{m-1}}(t_{m-1}) \Rightarrow \ldots \Rightarrow \forall^r t_0(RC_{T_0}(t_0) \Rightarrow RC_U(M[0 \mapsto \langle t_0, \ldots, t_{m-1}\rangle]))\ldots))\ldots)$$

$\mathcal{R}ed\mathcal{C}and\,(\Downarrow)$

$$\texttt{normrc} = \langle\langle \lambda\pi x.x\,\mathtt{z}\,*, \lambda tx.x\rangle\,, \lambda tu\pi xy.x\,(\lambda i.y\,(\mathtt{s}\,i))\rangle$$

$$[\![\texttt{normrc}]\!] \in |\mathcal{R}ed\mathcal{C}and\,(\Downarrow)|$$

$\mathcal{R}ed\mathcal{C}and\,(RC_T)$

For T type of system F built from variables X of the logic we define:

$$\mathtt{isrc}_T = \left\langle \left\langle \mathtt{isrc}_T^{(1)}, \mathtt{isrc}_T^{(2)} \right\rangle, \mathtt{isrc}_T^{(3)} \right\rangle$$

such that $\mathsf{FV}\,(\mathtt{isrc}_T) = \{x_X \mid X \in \mathsf{FV}\,(T)\}$

$\mathcal{R}ed\mathcal{C}and\,(RC_T)$

For T type of system F built from variables X of the logic we define:

$$\mathtt{isrc}_T = \left\langle \left\langle \mathtt{isrc}^{(1)}_T, \mathtt{isrc}^{(2)}_T \right\rangle, \mathtt{isrc}^{(3)}_T \right\rangle$$

such that $\mathrm{FV}\,(\mathtt{isrc}_T) = \{x_X \mid X \in \mathrm{FV}\,(T)\}$

$$\mathtt{isrc}^{(1)}_X = \mathrm{p}_1\,(\mathrm{p}_1\,x_X) \qquad \mathtt{isrc}^{(2)}_X = \mathrm{p}_2\,(\mathrm{p}_1\,x_X) \qquad \mathtt{isrc}^{(3)}_X = \mathrm{p}_3\,x_X \qquad \mathtt{isrc}^{(1)}_{T\to U} = \lambda\pi t x.\mathtt{isrc}^{(1)}_U\,(\mathtt{cons}\,\pi\,t)$$

$$\mathtt{isrc}^{(2)}_{T\to U} = \lambda t x.\mathtt{isrc}^{(2)}_U\,(\mathtt{app}\,t\,(\mathtt{var}\,\mathtt{z}))\left(x\,(\mathtt{var}\,\mathtt{z})\left(\mathtt{isrc}^{(1)}_T\,\mathtt{nil}\right)\right) \qquad \mathtt{isrc}^{(3)}_{T\to U} = \lambda t u \pi x v y.\mathtt{isrc}^{(3)}_U\;t\,u\,(\mathtt{cons}\,\pi\,v)\,(x\,v\,y)$$

$$\mathtt{isrc}^{(1)}_{\forall X\;T} = \lambda\pi x_X.\mathtt{isrc}^{(1)}_T\,\pi \qquad \mathtt{isrc}^{(3)}_{\forall X\;T} = \lambda t u \pi y x_X.\mathtt{isrc}^{(3)}_T\,(y\,x_X)$$

$$\mathtt{isrc}^{(2)}_{\forall X\;T} = \lambda t x.\mathtt{elim}_{\overline{X}\mapsto\mathcal{R}ed\mathcal{C}and\left(\overline{X}\right)\Rightarrow\forall^r t(RC_T(t)\Rightarrow t\downarrow),\Downarrow}\left(\lambda x_X \mathtt{isrc}^{(2)}_T\right)\mathtt{normrc}\;t\left(\mathtt{elim}_{\overline{X}\mapsto\mathcal{R}ed\mathcal{C}and\left(\overline{X}\right)\Rightarrow RC_T(t),\Downarrow}\;x\,\mathtt{normrc}\right)$$

$\mathcal{R}ed\mathcal{C}and\,(RC_T)$

For T type of system F built from variables X of the logic we define:

$$\mathtt{isrc}_T = \left\langle \left\langle \mathtt{isrc}^{(1)}_T, \mathtt{isrc}^{(2)}_T \right\rangle, \mathtt{isrc}^{(3)}_T \right\rangle$$

such that $\mathrm{FV}\,(\mathtt{isrc}_T) = \{x_X \mid X \in \mathrm{FV}\,(T)\}$

$$\mathtt{isrc}^{(1)}_X = \mathrm{p}_1\,(\mathrm{p}_1\,x_X) \qquad \mathtt{isrc}^{(2)}_X = \mathrm{p}_2\,(\mathrm{p}_1\,x_X) \qquad \mathtt{isrc}^{(3)}_X = \mathrm{p}_3\,x_X \qquad \mathtt{isrc}^{(1)}_{T\to U} = \lambda\pi t x.\mathtt{isrc}^{(1)}_U\,(\mathtt{cons}\,\pi\,t)$$

$$\mathtt{isrc}^{(2)}_{T\to U} = \lambda t x.\mathtt{isrc}^{(2)}_U\,(\mathtt{app}\,t\,(\mathtt{var}\,\mathtt{z}))\left(x\,(\mathtt{var}\,\mathtt{z})\left(\mathtt{isrc}^{(1)}_T\,\mathtt{nil}\right)\right) \qquad \mathtt{isrc}^{(3)}_{T\to U} = \lambda t u\pi x v y.\mathtt{isrc}^{(3)}_U\,t\,u\,(\mathtt{cons}\,\pi\,v)\,(x\,v\,y)$$

$$\mathtt{isrc}^{(1)}_{\forall X\,T} = \lambda\pi x_X.\mathtt{isrc}^{(1)}_T\,\pi \qquad \mathtt{isrc}^{(3)}_{\forall X\,T} = \lambda t u\pi y x_X.\mathtt{isrc}^{(3)}_T\,(y\,x_X)$$

$$\mathtt{isrc}^{(2)}_{\forall X\,T} = \lambda t x.\mathtt{elim}_{\overline{X}\mapsto\mathcal{R}ed\mathcal{C}and\left(\overline{X}\right)\Rightarrow\forall^r t(RC_T(t)\Rightarrow t\downarrow),\Downarrow}\left(\lambda x_X\mathtt{isrc}^{(2)}_T\right)\mathtt{normrc}\,t\left(\mathtt{elim}_{\overline{X}\mapsto\mathcal{R}ed\mathcal{C}and\left(\overline{X}\right)\Rightarrow RC_T(t),\Downarrow}\,x\,\mathtt{normrc}\right)$$

If $\vec{\mathfrak{X}} \in \mathcal{P}\,(\vec{\Lambda})$ and $\vec{\varphi} \in \left|\vec{\mathcal{R}ed\mathcal{C}and}\,\left(\overline{\mathfrak{X}}\right)\right|$ then:

$$[\![\mathtt{isrc}_T\,[\vec{\varphi}/\vec{x_X}]]\!] \in \left|\mathcal{R}ed\mathcal{C}and\,(RC_T)\left[\vec{\mathfrak{X}}/\vec{X}\right]\right|$$

$RC_T\left(M\left[0 \mapsto \langle t_0, \ldots, t_{m-1}\rangle\right]\right)$

If $\dfrac{\vdots}{\Gamma \vdash M : T}$ is a valid typing derivation in system F, define:

$$\texttt{adeq}_{\Gamma \vdash M:T}$$

such that $\mathrm{FV}\left(\texttt{adeq}_{\Gamma \vdash M:T}\right) = \{x_X \mid X \in \mathrm{FV}(\Gamma, T)\} \cup \{t_U \mid U \in \Gamma\} \cup \{y_U \mid U \in \Gamma\}$

$RC_T\left(M\left[0 \mapsto \langle t_0, \ldots, t_{m-1}\rangle\right]\right)$

If $\dfrac{\vdots}{\Gamma \vdash M : T}$ is a valid typing derivation in system F, define:

$$\mathtt{adeq}_{\Gamma\vdash M:T}$$

such that $FV\left(\mathtt{adeq}_{\Gamma\vdash M:T}\right) = \{x_X \mid X \in FV(\Gamma, T)\} \cup \{t_U \mid U \in \Gamma\} \cup \{y_U \mid U \in \Gamma\}$

$$\mathtt{adeq}_{\Gamma\vdash \underline{m}:U} = y_U \qquad \mathtt{adeq}_{\Gamma\vdash \lambda.M:U\to T} = \lambda t_U y_U.\mathtt{isrc}^{(3)}_T\left(M^*\left[\mathtt{s}\,\mathtt{z} \mapsto\uparrow t_\Gamma\right]\right) t_U\, \mathtt{nil}\, \mathtt{adeq}_{\Gamma,U\vdash M:T}$$

$$\mathtt{adeq}_{\Gamma\vdash M\,N:T} = \mathtt{adeq}_{\Gamma\vdash M:U\to T}\left(N^*\left[\mathtt{z} \mapsto t_\Gamma\right]\right) \mathtt{adeq}_{\Gamma\vdash N:U} \qquad \mathtt{adeq}_{\Gamma\vdash M:\forall X\,T} = \lambda x_X.\mathtt{adeq}_{\Gamma\vdash M:T}$$

$$\mathtt{adeq}_{\Gamma\vdash M:T\{U/X\}} = \mathtt{elim}_{\overline{X}\mapsto \mathcal{R}ed\mathcal{C}and\left(\overline{X}\right)\Rightarrow RC_T(M[0\mapsto t_\Gamma]),RC_U}\, \mathtt{adeq}_{\Gamma\vdash M:\forall X\,T}\, \mathtt{isrc}_U$$

$RC_T\left(M\left[0 \mapsto \langle t_0, \ldots, t_{m-1}\rangle\right]\right)$

If $\dfrac{\vdots}{\Gamma \vdash M : T}$ is a valid typing derivation in system F, define:

$$\mathtt{adeq}_{\Gamma \vdash M:T}$$

such that $\mathrm{FV}\left(\mathtt{adeq}_{\Gamma \vdash M:T}\right) = \{x_X \mid X \in \mathrm{FV}(\Gamma, T)\} \cup \{t_U \mid U \in \Gamma\} \cup \{y_U \mid U \in \Gamma\}$

$$\mathtt{adeq}_{\Gamma \vdash \underline{m}:U} = y_U \qquad \mathtt{adeq}_{\Gamma \vdash \lambda.M:U \to T} = \lambda t_U y_U.\mathtt{isrc}^{(3)}_T\left(M^*\left[\mathtt{s}\,\mathtt{z} \mapsto \uparrow t_\Gamma\right]\right) t_U\,\mathtt{nil}\,\mathtt{adeq}_{\Gamma, U \vdash M:T}$$

$$\mathtt{adeq}_{\Gamma \vdash M\,N:T} = \mathtt{adeq}_{\Gamma \vdash M:U \to T}\left(N^*\left[\mathtt{z} \mapsto t_\Gamma\right]\right)\mathtt{adeq}_{\Gamma \vdash N:U} \qquad \mathtt{adeq}_{\Gamma \vdash M:\forall X\,T} = \lambda x_X.\mathtt{adeq}_{\Gamma \vdash M:T}$$

$$\mathtt{adeq}_{\Gamma \vdash M:T\{U/X\}} = \mathtt{elim}_{\overline{X} \mapsto \mathcal{R}ed\mathcal{C}and\left(\overline{X}\right) \Rightarrow RC_T(M[0 \mapsto t_\Gamma]), RC_U}\,\mathtt{adeq}_{\Gamma \vdash M:\forall X\,T}\,\mathtt{isrc}_U$$

If $\vec{\mathfrak{X}} \in \vec{\mathcal{P}(\Lambda)}$, $\vec{\varphi} \in \left|\vec{\mathcal{R}ed\mathcal{C}and}\left(\overline{\mathfrak{X}}\right)\right|$, $\mathfrak{M}_U \in \Lambda$ and $\psi_U \in \left|RC_U\left(\mathfrak{M}_U\right)\left[\vec{\mathfrak{X}}/\vec{X}\right]\right|$ for $U \in \Gamma$, then:

$$\left[\left[\mathtt{adeq}_{\Gamma \vdash M:T}\left[\vec{\mathfrak{M}}_U/\vec{t_U}, \vec{\psi}_U/\vec{y_U}\right]\right]\right] \in \left|RC_T\left(M\left[0 \mapsto \vec{\mathfrak{M}}_U\right]\right)\left[\vec{\mathfrak{X}}/\vec{X}\right]\right|$$

The translation of $M : T$

Extracting the bound

In particular if M closed term of closed type T, then:

$$[\![\mathtt{adeq}_{\vdash M:T}]\!] \in |RC_T(M)| \;\; \texttt{and} \;\; \left[\!\!\left[\mathtt{isrc}_T^{(2)}\right]\!\!\right] \in |\forall^r t\,(RC_T(t) \Rightarrow t\!\downarrow)|$$

Extracting the bound

In particular if M closed term of closed type T, then:

$$[\![\texttt{adeq}_{\vdash M:T}]\!] \in |RC_T(M)| \text{ and } \left[\!\!\left[\texttt{isrc}_T^{(2)}\right]\!\!\right] \in |\forall^r t\,(RC_T(t) \Rightarrow t\!\downarrow)|$$

therefore:

$$\left[\!\!\left[\texttt{isrc}_T^{(2)}\, M^* \,\texttt{adeq}_{\vdash M:T}\right]\!\!\right] \in |M\!\downarrow|$$

Extracting the bound

In particular if M closed term of closed type T, then:

$$[\![\texttt{adeq}_{\vdash M:T}]\!] \in |RC_T(M)| \text{ and } \left[\!\left[\texttt{isrc}_T^{(2)}\right]\!\right] \in |\forall^r t\,(RC_T(t) \Rightarrow t{\downarrow})|$$

therefore:

$$\left[\!\left[\texttt{isrc}_T^{(2)}\, M^*\, \texttt{adeq}_{\vdash M:T}\right]\!\right] \in |M{\downarrow}|$$

recall that $M{\downarrow} \equiv \neg\forall^r i\,\neg M{\downarrow}^i$. Fix now:

$$\perp\!\!\!\perp = \{\mathfrak{n} \in \mathbb{N} \,|\, M \text{ normalizes in at most } \mathfrak{n} \text{ steps}\}$$

Extracting the bound

In particular if M closed term of closed type T, then:

$$[\![\mathtt{adeq}_{\vdash M:T}]\!] \in |RC_T(M)| \text{ and } \left[\!\!\left[\mathtt{isrc}_T^{(2)}\right]\!\!\right] \in |\forall^r t\,(RC_T(t) \Rightarrow t\!\downarrow)|$$

therefore:

$$\left[\!\!\left[\mathtt{isrc}_T^{(2)}\, M^*\, \mathtt{adeq}_{\vdash M:T}\right]\!\!\right] \in |M\!\downarrow|$$

recall that $M\!\downarrow\, \equiv \neg\forall^r i\, \neg M\!\downarrow^i$. Fix now:

$$\perp\!\!\!\perp = \{\mathfrak{n} \in \mathbb{N} \mid M \text{ normalizes in at most } \mathfrak{n} \text{ steps}\}$$

By case disjunction, $[\![\lambda x_{\_}.x]\!] \in \left|\forall^r i\, \neg M\!\downarrow^i\right|$ and therefore:

$$\left[\!\!\left[\mathtt{isrc}_T^{(2)}\, M^*\, \mathtt{adeq}_{\vdash M:T}\,(\lambda x_{\_}.x)\right]\!\!\right] \in |f\!\!f| = \perp\!\!\!\perp$$

Extracting the bound

In particular if M closed term of closed type T, then:

$$[\![\mathtt{adeq}_{\vdash M:T}]\!] \in |RC_T(M)| \text{ and } \left[\!\!\left[\mathtt{isrc}^{(2)}_T\right]\!\!\right] \in |\forall^r t\,(RC_T(t) \Rightarrow t\!\downarrow)|$$

therefore:

$$\left[\!\!\left[\mathtt{isrc}^{(2)}_T\, M^*\,\mathtt{adeq}_{\vdash M:T}\right]\!\!\right] \in |M\!\downarrow|$$

recall that $M\!\downarrow\; \equiv \neg\forall^r i\,\neg M\!\downarrow^i$. Fix now:

$$\perp\!\!\!\perp = \{\mathfrak{n} \in \mathbb{N} \mid M \text{ normalizes in at most } \mathfrak{n} \text{ steps}\}$$

By case disjunction, $[\![\lambda x\_.x]\!] \in \left|\forall^r i\,\neg M\!\downarrow^i\right|$ and therefore:

$$\left[\!\!\left[\mathtt{isrc}^{(2)}_T\, M^*\,\mathtt{adeq}_{\vdash M:T}\,(\lambda x\_.x)\right]\!\!\right] \in |f\!f| = \perp\!\!\!\perp$$

so by computational adequacy:

$$\mathtt{isrc}^{(2)}_T\, M^*\,\mathtt{adeq}_{\vdash M:T}\,(\lambda x\_.x) \rightsquigarrow^* \mathtt{s}^{\mathfrak{n}}\,\mathtt{z}$$

where $\mathfrak{n}$ is such that M normalizes in at most $\mathfrak{n}$ steps

Computing the normal form

$$\mathtt{isrc}_T^{(2)}\, M^*\, \mathtt{adeq}_{\vdash M:T}\, (\lambda x\_.x) \rightsquigarrow^* \mathtt{s}^{\mathfrak{n}}\, \mathtt{z}$$

where $\mathfrak{n}$ is such that M normalizes in at most $\mathfrak{n}$ steps

Computing the normal form

$$\mathtt{isrc}_T^{(2)}\, M^*\, \mathtt{adeq}_{\vdash M:T}\,(\lambda x_{\_}.x) \rightsquigarrow^* \mathtt{s}^{\mathfrak{n}}\,\mathtt{z}$$

where $\mathfrak{n}$ is such that M normalizes in at most $\mathfrak{n}$ steps

We can easily define a term $\mathtt{red} : \lambda \to \lambda$ in system ΛT_{br} such that:

- if $M \succ N$ then $\mathtt{red}\, M^* \rightsquigarrow^* N^*$
- if M is in weak head normal form then $\mathtt{red}\, M^* \rightsquigarrow^* M^*$

Computing the normal form

$$\mathtt{isrc}_T^{(2)}\, M^*\, \mathtt{adeq}_{\vdash M : T}\, (\lambda x_{\_}.x) \rightsquigarrow^* \mathtt{s}^{\mathfrak{n}}\, \mathtt{z}$$

where $\mathfrak{n}$ is such that M normalizes in at most $\mathfrak{n}$ steps

We can easily define a term $\mathtt{red} : \lambda \to \lambda$ in system ΛT_{br} such that:

- if $M \succ N$ then $\mathtt{red}\, M^* \rightsquigarrow^* N^*$
- if M is in weak head normal form then $\mathtt{red}\, M^* \rightsquigarrow^* M^*$

therefore:

$$\mathtt{it}_\iota\, M^*\, \mathtt{red}\left(\mathtt{isrc}_T^{(2)}\, M^*\, \mathtt{adeq}_{\vdash M : T}\, (\lambda x_{\_}.x)\right) \rightsquigarrow^* N^*$$

where N is the weak head normal form of M

Conclusion

Conclusion

- Translation of system F into a simply-typed total programming language

Conclusion

- Translation of system F into a simply-typed total programming language
- No more impredicativity?

Conclusion

- Translation of system F into a simply-typed total programming language
- No more impredicativity?
- Normalization of system F reduced to Zorn's lemma?

Conclusion

- Translation of system F into a simply-typed total programming language
- No more impredicativity?
- Normalization of system F reduced to Zorn's lemma?
- Weak form of bar recursion: comprehension scheme weaker than classical countable choice?

Conclusion

- Translation of system F into a simply-typed total programming language
- No more impredicativity?
- Normalization of system F reduced to Zorn's lemma?
- Weak form of bar recursion: comprehension scheme weaker than classical countable choice?

Many possible improvements:

Conclusion

- Translation of system F into a simply-typed total programming language
- No more impredicativity?
- Normalization of system F reduced to Zorn's lemma?
- Weak form of bar recursion: comprehension scheme weaker than classical countable choice?

Many possible improvements:

- Extract directly the normal form rather than a bound
  - Normal form of M from normal form of $M\,\underline{0}$
  - Head reduction rather than weak head reduction

Conclusion

- Translation of system F into a simply-typed total programming language
- No more impredicativity?
- Normalization of system F reduced to Zorn's lemma?
- Weak form of bar recursion: comprehension scheme weaker than classical countable choice?

Many possible improvements:

- Extract directly the normal form rather than a bound
  - Normal form of M from normal form of $M\,\underline{0}$
  - Head reduction rather than weak head reduction
- Strong normalization
  - Realizers much more complicated
  - But we get the β-normal form

Conclusion

- Translation of system F into a simply-typed total programming language
- No more impredicativity?
- Normalization of system F reduced to Zorn's lemma?
- Weak form of bar recursion: comprehension scheme weaker than classical countable choice?

Many possible improvements:

- Extract directly the normal form rather than a bound
  - Normal form of M from normal form of $M\,\underline{0}$
  - Head reduction rather than weak head reduction
- Strong normalization
  - Realizers much more complicated
  - But we get the β-normal form

Implementation of the translation

Adequacy of bar recursion: preliminaries

$$\llbracket \lambda xy.\mathtt{brec}\,(\lambda z.\mathtt{exf}_{A^-}\,(x\,z))\,y\,\{\}\rrbracket \in \left|\forall t\,\exists b\,A^-\,(b,t) \Rightarrow \exists X\,\forall^r t\,A^-\,(t \in X, t)\right|$$

Adequacy of bar recursion: preliminaries

$$\llbracket \lambda xy.\mathtt{brec}\,(\lambda z.\mathtt{exf}_{A^-}\,(x\,z))\,y\,\{\}\rrbracket \in \left|\forall t\,\exists b\,A^-\,(b,t) \Rightarrow \exists X\,\forall^r t\,A^-\,(t \in X, t)\right|$$

Let $\varphi \in |\forall t\,\exists b\,A^-\,(b,t)|$ and $\psi \in |\forall X\,\neg\forall^r t\,A^-\,(t \in X, t)|$, and write $\theta \triangleq \llbracket \mathtt{brec}\,(\lambda z.\mathtt{exf}_{A^-}\,(\varphi\,z))\,\psi\rrbracket$.

Adequacy of bar recursion: preliminaries

$$[\![\lambda x y.\texttt{brec}\,(\lambda z.\texttt{exf}_{A^-}\,(x\,z))\,y\,\{\}]\!] \in \left|\forall t\,\exists b\,A^-\,(b,t) \Rightarrow \exists X\,\forall^r t\,A^-\,(t \in X, t)\right|$$

Let $\varphi \in |\forall t\,\exists b\,A^-\,(b,t)|$ and $\psi \in |\forall X\,\neg\forall^r t\,A^-\,(t \in X, t)|$, and write $\theta \triangleq [\![\texttt{brec}\,(\lambda z.\texttt{exf}_{A^-}\,(\varphi\,z))\,\psi]\!]$. $[\![\theta\ \{\}]\!] \in |\mathit{ff}|$?

Adequacy of bar recursion: preliminaries

$$[\![\lambda xy.\texttt{brec}\,(\lambda z.\texttt{exf}_{A^-}\,(x\,z))\,y\,\{\}]\!] \in \left|\forall t\,\exists b\,A^-\,(b,t) \Rightarrow \exists X\,\forall^r t\,A^-\,(t \in X, t)\right|$$

Let $\varphi \in |\forall t\,\exists b\,A^-\,(b,t)|$ and $\psi \in |\forall X\,\neg\forall^r t\,A^-\,(t \in X, t)|$, and write $\theta \triangleq [\![\texttt{brec}\,(\lambda z.\texttt{exf}_{A^-}\,(\varphi\,z))\,\psi]\!]$. $[\![\theta\ \{\}]\!] \in |\mathit{ff}|$?

Let E be the set of $\xi \in \left[\!\left[A^{-*^\dagger}\right]\!\right]$ such that:

- $\pi_2\,(\xi\,(\mathfrak{M})) \in |A^-\,(\mathfrak{tt},\mathfrak{M})| \cup |A^-\,(\mathfrak{ff},\mathfrak{M})|$ if $\pi_1\,(\xi\,(\mathfrak{M})) = 0$
- $\xi\,(\mathfrak{M}) = (1, [\![\texttt{can}_{A^{-*}}]\!])$ otherwise
- $\xi\,(\perp) = \perp$
- $\theta\,(\xi) \notin |\mathit{ff}|$

Adequacy of bar recursion: preliminaries

$$[\![\lambda xy.\texttt{brec}\,(\lambda z.\texttt{exf}_{A^-}\,(x\,z))\,y\,\{\}]\!] \in \left|\forall t\,\exists b\,A^-\,(b,t) \Rightarrow \exists X\,\forall^r t\,A^-\,(t\in X,t)\right|$$

Let $\varphi \in |\forall t\,\exists b\,A^-\,(b,t)|$ and $\psi \in |\forall X\,\neg\forall^r t\,A^-\,(t\in X,t)|$, and write $\theta \triangleq [\![\texttt{brec}\,(\lambda z.\texttt{exf}_{A^-}\,(\varphi\,z))\,\psi]\!]$. $[\![\theta\ \{\}]\!] \in |\mathit{ff}|$?

Let E be the set of $\xi \in \left[\!\left[A^{-*^\dagger}\right]\!\right]$ such that:

- $\pi_2\,(\xi\,(\mathfrak{M})) \in |A^-\,(\mathfrak{tt},\mathfrak{M})| \cup |A^-\,(\mathfrak{ff},\mathfrak{M})|$ if $\pi_1\,(\xi\,(\mathfrak{M})) = 0$
- $\xi\,(\mathfrak{M}) = (1, [\![\texttt{can}_{A^{-*}}]\!])$ otherwise
- $\xi\,(\bot) = \bot$
- $\theta\,(\xi) \notin |\mathit{ff}|$

and let $\prec$ be the following partial order on E:

$$\xi \prec \xi' \quad\Longleftrightarrow\quad \left(\pi_1\,(\xi\,(\mathfrak{M})) = 0 \Rightarrow \xi'\,(\mathfrak{M}) = \xi\,(\mathfrak{M})\right)$$

$$[\![\theta\ \{\}]\!] \in |\mathit{ff}| \Longleftrightarrow [\![\{\}]\!] \notin E$$

Adequacy of bar recursion: Zorn's lemma

Theorem (Zorn's lemma on $(E, \prec)$)

if every chain (totally ordered subset) of E has an upper bound in E, then E has a maximal element

Adequacy of bar recursion: Zorn's lemma

Theorem (Zorn's lemma on $(E, \prec)$)

if every chain (totally ordered subset) of E has an upper bound in E, then E has a maximal element

We prove two things:

- Every non-empty chain has an upper bound
- E has no maximal element

Adequacy of bar recursion: Zorn's lemma

Theorem (Zorn's lemma on $(E, \prec)$)

if every chain (totally ordered subset) of E has an upper bound in E, then E has a maximal element

We prove two things:

- Every non-empty chain has an upper bound
- E has no maximal element

Therefore the empty chain has no upper bound, i.e. $E = \emptyset$. In particular $[\![\{\}]\!] \notin E$, we are done.

Adequacy of bar recursion: chains $\neq \emptyset$ have upper bounds

- C non-empty chain

Adequacy of bar recursion: chains $\neq \emptyset$ have upper bounds

- C non-empty chain
- $\xi_{max}(\mathfrak{M}) = \begin{cases} \xi(\mathfrak{M}) & \text{if } \pi_1(\xi(\mathfrak{M})) = 0 \text{ for some } \xi \in C \\ (1, [\![\mathtt{can}_{A^{-*}}]\!]) & \text{otherwise} \end{cases}$

  $\xi_{max}(\bot) = \bot$

Adequacy of bar recursion: chains $\neq \emptyset$ have upper bounds

- C non-empty chain
- $\xi_{max}(\mathfrak{M}) = \begin{cases} \xi(\mathfrak{M}) & \text{if } \pi_1(\xi(\mathfrak{M})) = 0 \text{ for some } \xi \in C \\ (1, [\![\texttt{can}_{A^{-*}}]\!]) & \text{otherwise} \end{cases}$

  $\xi_{max}(\bot) = \bot$
- $\xi_{max} \in E$: we want to prove $\theta(\xi_{max}) \notin |\mathit{ff}|$.
  - Suppose $\theta(\xi_{max}) \in |\mathit{ff}|$

Adequacy of bar recursion: chains $\neq \emptyset$ have upper bounds

- C non-empty chain
- $\xi_{max}(\mathfrak{M}) = \begin{cases} \xi(\mathfrak{M}) & \text{if } \pi_1(\xi(\mathfrak{M})) = 0 \text{ for some } \xi \in C \\ (1, [\![\mathtt{can}_{A^{-*}}]\!]) & \text{otherwise} \end{cases}$

  $\xi_{max}(\bot) = \bot$
- $\xi_{max} \in E$: we want to prove $\theta(\xi_{max}) \notin |\mathit{ff}|$.
  - Suppose $\theta(\xi_{max}) \in |\mathit{ff}|$
  - Continuity of θ implies existence of finite $F \subseteq \Lambda$ such that:

$$\forall \xi \left(\forall \mathfrak{M} \in F \left(\xi(\mathfrak{M}) = \xi_{max}(\mathfrak{M}) \right) \Rightarrow \theta(\xi) = \theta(\xi_{max}) \right)$$

Adequacy of bar recursion: chains $\neq \emptyset$ have upper bounds

- C non-empty chain
- $\xi_{max}(\mathfrak{M}) = \begin{cases} \xi(\mathfrak{M}) \text{ if } \pi_1(\xi(\mathfrak{M})) = 0 \text{ for some } \xi \in C \\ (1, [\![\mathtt{can}_{A^{-*}}]\!]) \text{ otherwise} \end{cases}$

  $\xi_{max}(\bot) = \bot$
- $\xi_{max} \in E$: we want to prove $\theta(\xi_{max}) \notin |\mathit{ff}|$.
  - Suppose $\theta(\xi_{max}) \in |\mathit{ff}|$
  - Continuity of θ implies existence of finite $F \subseteq \Lambda$ such that:

    $$\forall \xi (\forall \mathfrak{M} \in F (\xi(\mathfrak{M}) = \xi_{max}(\mathfrak{M})) \Rightarrow \theta(\xi) = \theta(\xi_{max}))$$

  - Since C is a non-empty chain there exists $\xi \in C$ such that $\forall \mathfrak{M} \in F (\xi(\mathfrak{M}) = \xi_{max}(\mathfrak{M}))$ and therefore $\theta(\xi) = \theta(\xi_{max})$. $\xi \in E$ so $\theta(\xi) \notin |\mathit{ff}|$ and $\theta(\xi_{max}) = \theta(\xi) \notin |\mathit{ff}|$, contradiction.

Adequacy of bar recursion: chains $\neq \emptyset$ have upper bounds

- C non-empty chain
- $\xi_{max}(\mathfrak{M}) = \begin{cases} \xi(\mathfrak{M}) & \text{if } \pi_1(\xi(\mathfrak{M})) = 0 \text{ for some } \xi \in C \\ (1, [\![\mathtt{can}_{A^{-*}}]\!]) & \text{otherwise} \end{cases}$

  $\xi_{max}(\bot) = \bot$
- $\xi_{max} \in E$: we want to prove $\theta(\xi_{max}) \notin |\mathit{ff}|$.
  - Suppose $\theta(\xi_{max}) \in |\mathit{ff}|$
  - Continuity of θ implies existence of finite $F \subseteq \Lambda$ such that:

    $$\forall \xi (\forall \mathfrak{M} \in F (\xi(\mathfrak{M}) = \xi_{max}(\mathfrak{M})) \Rightarrow \theta(\xi) = \theta(\xi_{max}))$$

  - Since C is a non-empty chain there exists $\xi \in C$ such that $\forall \mathfrak{M} \in F (\xi(\mathfrak{M}) = \xi_{max}(\mathfrak{M}))$ and therefore $\theta(\xi) = \theta(\xi_{max})$. $\xi \in E$ so $\theta(\xi) \notin |\mathit{ff}|$ and $\theta(\xi_{max}) = \theta(\xi) \notin |\mathit{ff}|$, contradiction.

Therefore $\xi_{max} \in E$ is an upper bound for C

Adequacy of bar recursion: E has no maximal element

- Suppose ξ is a maximal element of E

Adequacy of bar recursion: E has no maximal element

- Suppose ξ is a maximal element of E
- $[\![\theta\,\xi]\!] = [\![\psi\,(\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\}))))]\!]$

Adequacy of bar recursion: E has no maximal element

- Suppose ξ is a maximal element of E
- $[\![\theta\,\xi]\!] = [\![\psi\,(\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\}))))]\!]$
- Let $\mathfrak{X} = \{\mathfrak{M} \in \Lambda \mid \pi_2\,(\xi\,(\mathfrak{M})) \in |A^-\,(\mathfrak{tt}, \mathfrak{M})|\}$

Adequacy of bar recursion: E has no maximal element

- Suppose ξ is a maximal element of E
- $[\![\theta\,\xi]\!] = [\![\psi\,(\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\}))))]\!]$
- Let $\mathfrak{X} = \{\mathfrak{M} \in \Lambda \mid \pi_2\,(\xi\,(\mathfrak{M})) \in |A^-\,(\mathtt{tt}, \mathfrak{M})|\}$
- $\psi \in |\neg\forall^r t\, A^-\,(t \in \mathfrak{X}, t)|$ and $[\![\theta\,\xi]\!] = \theta\,(\xi) \notin |\mathit{ff}|$ so:

  $[\![\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\})))]\!] \notin \left|\forall^r t\, A^-\,(t \in \mathfrak{X}, t)\right|$

Adequacy of bar recursion: E has no maximal element

- Suppose ξ is a maximal element of E
- $[\![\theta\,\xi]\!] = [\![\psi\,(\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\}))))]\!]$
- Let $\mathfrak{X} = \{\mathfrak{M} \in \Lambda \mid \pi_2\,(\xi\,(\mathfrak{M})) \in |A^-\,(\mathbb{tt}, \mathfrak{M})|\}$
- $\psi \in |\neg\forall^r t\, A^-\,(t \in \mathfrak{X}, t)|$ and $[\![\theta\,\xi]\!] = \theta\,(\xi) \notin |\mathit{ff}|$ so:

  $[\![\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\})))]\!] \notin \left|\forall^r t\, A^-\,(t \in \mathfrak{X}, t)\right|$
- there is some $\mathfrak{M} \in \Lambda$ such that:

  $[\![(\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\}))))\,\mathfrak{M}]\!] \notin \left|A^-\,(\mathfrak{M} \in \mathfrak{X}, \mathfrak{M})\right|$

Adequacy of bar recursion: E has no maximal element

- Suppose ξ is a maximal element of E
- $[\![\theta\,\xi]\!] = [\![\psi\,(\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\}))))]\!]$
- Let $\mathfrak{X} = \{\mathfrak{M} \in \Lambda \mid \pi_2\,(\xi\,(\mathfrak{M})) \in |A^-\,(\mathtt{tt}, \mathfrak{M})|\}$
- $\psi \in |\neg\forall^r t\, A^-\,(t \in \mathfrak{X}, t)|$ and $[\![\theta\,\xi]\!] = \theta\,(\xi) \notin |\mathit{ff}|$ so:

  $$[\![\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\})))]\!] \notin \left|\forall^r t\, A^-\,(t \in \mathfrak{X}, t)\right|$$

- there is some $\mathfrak{M} \in \Lambda$ such that:

  $$[\![(\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\}))))\,\mathfrak{M}]\!] \notin \left|A^-\,(\mathfrak{M} \in \mathfrak{X}, \mathfrak{M})\right|$$

  - If $\pi_1\,(\xi\,(\mathfrak{M})) = 0$ then $\pi_2\,(\xi\,(\mathfrak{M})) \notin |A^-\,(\mathfrak{M} \in \mathfrak{X}, \mathfrak{M})|$, absurd by definition of $\mathfrak{X}$ since $\xi \in E$

Adequacy of bar recursion: E has no maximal element

- Suppose ξ is a maximal element of E
- $[\![\theta\,\xi]\!] = [\![\psi\,(\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\}))))]\!]$
- Let $\mathfrak{X} = \{\mathfrak{M} \in \Lambda \mid \pi_2\,(\xi\,(\mathfrak{M})) \in |A^-\,(\mathtt{tt}, \mathfrak{M})|\}$
- $\psi \in |\neg\forall^r t\, A^-\,(t \in \mathfrak{X}, t)|$ and $[\![\theta\,\xi]\!] = \theta\,(\xi) \notin |\mathit{ff}|$ so:

  $$[\![\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\})))]\!] \notin \left|\forall^r t\, A^-\,(t \in \mathfrak{X}, t)\right|$$

- there is some $\mathfrak{M} \in \Lambda$ such that:

  $$[\![(\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\}))))\,\mathfrak{M}]\!] \notin \left|A^-\,(\mathfrak{M} \in \mathfrak{X}, \mathfrak{M})\right|$$

  - If $\pi_1\,(\xi\,(\mathfrak{M})) = 0$ then $\pi_2\,(\xi\,(\mathfrak{M})) \notin |A^-\,(\mathfrak{M} \in \mathfrak{X}, \mathfrak{M})|$, absurd by definition of $\mathfrak{X}$ since $\xi \in E$
  - Then $[\![\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{\mathfrak{M} \mapsto y\})))]\!] \notin |A^-\,(\mathfrak{M} \in \mathfrak{X}, \mathfrak{M})|$

Adequacy of bar recursion: E has no maximal element

- Suppose ξ is a maximal element of E
- $[\![\theta\,\xi]\!] = [\![\psi\,(\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\}))))]\!]$
- Let $\mathfrak{X} = \{\mathfrak{M} \in \Lambda \mid \pi_2\,(\xi\,(\mathfrak{M})) \in |A^-\,(\mathtt{tt}, \mathfrak{M})|\}$
- $\psi \in |\neg\forall^r t\, A^-\,(t \in \mathfrak{X}, t)|$ and $[\![\theta\,\xi]\!] = \theta\,(\xi) \notin |\mathit{ff}|$ so:

  $$[\![\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\})))]\!] \notin \left|\forall^r t\, A^-\,(t \in \mathfrak{X}, t)\right|$$

- there is some $\mathfrak{M} \in \Lambda$ such that:

  $$[\![(\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\}))))\,\mathfrak{M}]\!] \notin \left|A^-\,(\mathfrak{M} \in \mathfrak{X}, \mathfrak{M})\right|$$

  - If $\pi_1\,(\xi\,(\mathfrak{M})) = 0$ then $\pi_2\,(\xi\,(\mathfrak{M})) \notin |A^-\,(\mathfrak{M} \in \mathfrak{X}, \mathfrak{M})|$, absurd by definition of $\mathfrak{X}$ since $\xi \in E$
  - Then $[\![\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{\mathfrak{M} \mapsto y\})))]\!] \notin |A^-\,(\mathfrak{M} \in \mathfrak{X}, \mathfrak{M})|$
  - $[\![\varphi\,(\lambda y.\theta\,(\xi \cup \{\mathfrak{M} \mapsto y\}))]\!] \notin |\mathit{ff}|$ but $\varphi \in |\neg\forall b\,\neg A^-\,(b, \mathfrak{M})|$:

    $$[\![\lambda y.\theta\,(\xi \cup \{\mathfrak{M} \mapsto y\})]\!] \notin \left|\forall b\,\neg A^-\,(b, \mathfrak{M})\right|$$

Adequacy of bar recursion: E has no maximal element

- Suppose ξ is a maximal element of E
- $[\![\theta\,\xi]\!] = [\![\psi\,(\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\}))))]\!]$
- Let $\mathfrak{X} = \{\mathfrak{M} \in \Lambda \mid \pi_2\,(\xi\,(\mathfrak{M})) \in |A^-\,(\mathfrak{tt}, \mathfrak{M})|\}$
- $\psi \in |\neg\forall^r t\, A^-\,(t \in \mathfrak{X}, t)|$ and $[\![\theta\,\xi]\!] = \theta\,(\xi) \notin |\mathit{ff}|$ so:

  $$[\![\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\})))]\!] \notin \left|\forall^r t\, A^-\,(t \in \mathfrak{X}, t)\right|$$

- there is some $\mathfrak{M} \in \Lambda$ such that:

  $$[\![(\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\}))))\,\mathfrak{M}]\!] \notin \left|A^-\,(\mathfrak{M} \in \mathfrak{X}, \mathfrak{M})\right|$$

  - If $\pi_1\,(\xi\,(\mathfrak{M})) = 0$ then $\pi_2\,(\xi\,(\mathfrak{M})) \notin |A^-\,(\mathfrak{M} \in \mathfrak{X}, \mathfrak{M})|$, absurd by definition of $\mathfrak{X}$ since $\xi \in E$
  - Then $[\![\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{\mathfrak{M} \mapsto y\})))]\!] \notin |A^-\,(\mathfrak{M} \in \mathfrak{X}, \mathfrak{M})|$
  - $[\![\varphi\,(\lambda y.\theta\,(\xi \cup \{\mathfrak{M} \mapsto y\}))]\!] \notin |\mathit{ff}|$ but $\varphi \in |\neg\forall b\,\neg A^-\,(b, \mathfrak{M})|$:

    $$[\![\lambda y.\theta\,(\xi \cup \{\mathfrak{M} \mapsto y\})]\!] \notin \left|\forall b\,\neg A^-\,(b, \mathfrak{M})\right|$$

  - There exists $\zeta \in |A^-\,(\mathfrak{tt}, \mathfrak{M})| \cup |A^-\,(\mathfrak{ff}, \mathfrak{M})|$ such that:

    $$[\![\theta\,(\xi \cup \{\mathfrak{M} \mapsto \zeta\})]\!] \notin |\mathit{ff}|$$

Adequacy of bar recursion: E has no maximal element

- Suppose ξ is a maximal element of E
- $[\![\theta\,\xi]\!] = [\![\psi\,(\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\}))))]\!]$
- Let $\mathfrak{X} = \{\mathfrak{M} \in \Lambda \mid \pi_2\,(\xi\,(\mathfrak{M})) \in |A^-\,(\mathfrak{tt}, \mathfrak{M})|\}$
- $\psi \in |\neg\forall^r t\, A^-\,(t \in \mathfrak{X}, t)|$ and $[\![\theta\,\xi]\!] = \theta\,(\xi) \notin |\mathfrak{ff}|$ so:

  $$[\![\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\})))]\!] \notin \left|\forall^r t\, A^-\,(t \in \mathfrak{X}, t)\right|$$

- there is some $\mathfrak{M} \in \Lambda$ such that:

  $$[\![(\xi \mid \lambda x.\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{x \mapsto y\}))))\,\mathfrak{M}]\!] \notin \left|A^-\,(\mathfrak{M} \in \mathfrak{X}, \mathfrak{M})\right|$$

  - If $\pi_1\,(\xi\,(\mathfrak{M})) = 0$ then $\pi_2\,(\xi\,(\mathfrak{M})) \notin |A^-\,(\mathfrak{M} \in \mathfrak{X}, \mathfrak{M})|$, absurd by definition of $\mathfrak{X}$ since $\xi \in E$
  - Then $[\![\mathtt{exf}_{A^-}\,(\varphi\,(\lambda y.\theta\,(\xi \cup \{\mathfrak{M} \mapsto y\})))]\!] \notin |A^-\,(\mathfrak{M} \in \mathfrak{X}, \mathfrak{M})|$
  - $[\![\varphi\,(\lambda y.\theta\,(\xi \cup \{\mathfrak{M} \mapsto y\}))]\!] \notin |\mathfrak{ff}|$ but $\varphi \in |\neg\forall b\, \neg A^-\,(b, \mathfrak{M})|$:

    $$[\![\lambda y.\theta\,(\xi \cup \{\mathfrak{M} \mapsto y\})]\!] \notin \left|\forall b\, \neg A^-\,(b, \mathfrak{M})\right|$$

  - There exists $\zeta \in |A^-\,(\mathfrak{tt}, \mathfrak{M})| \cup |A^-\,(\mathfrak{ff}, \mathfrak{M})|$ such that:

    $$[\![\theta\,(\xi \cup \{\mathfrak{M} \mapsto \zeta\})]\!] \notin |\mathfrak{ff}|$$

  - $[\![\xi \cup \{\mathfrak{M} \mapsto \zeta\}]\!] \in E$ and $\xi \prec [\![\xi \cup \{\mathfrak{M} \mapsto \zeta\}]\!]$, contradiction