Advanced Type Systems, Lecture II

Encoding Zermelo's Set Theory into System $F\omega^2$ with Universes

Alexandre Miquel (miquel@cs.chalmers.se)

SUMMER 02: Proofs as Programs – Eugene, Oregon

Introduction

- **Background:** Encoding sets as highly-branching trees [P. Aczel, B. Werner]

$$\begin{array}{l} \text{Inductive set} : \text{Type}_2 := \\ \mid \ \text{node} : \Pi I : \text{Type}_1 \,.\, (I \to \text{set}) \to \text{set} \end{array}$$

  – Gives proof-theoretical strength of Martin-Löf type theory [P. Aczel]

  – Strongly relies on generalized induction

  $\Rightarrow$ This method can not be used in (impredicative) PTS

- **Sets as pointed graphs**

  – Ideas borrowed to non well-founded set theory [F. Honsell, P. Aczel]

  – Does not rely on inductive definitions

  – Relates impredicative PTS with Zermelo's Set Theory

The pure type system $F\omega^2$

Sorts:	Prop,	Type_i	$(i \geq 1)$
Axioms:	Prop : Type_1,	Type_i : Type_{i+1}	
Rules:	(Prop, Prop, Prop),	(Type_i, Prop, Prop),	(Type_i, Type_j, $\mathsf{Type}_{\max(i,j)}$)

Remark: $F\omega^2$ is a sub-system of the Calculus of Constructions with universes (i.e. $F\omega^2 \subset \mathsf{CC}_\omega \subset \mathsf{ECC}$) in which we forbid

– Any kind of cumulativity (i.e. the inclusions **Prop** $\subset$ Type_1 and $\mathsf{Type}_i \subset \mathsf{Type}_{i+1}$);

– Types depending on proofs (i.e. the rule (**Prop**, Type_i, Type_i))

Stratified presentation of $F\omega^2$

A well-typed term M of $F\omega^2$ is either:

- an (object) term: $M : T : \mathsf{Type}_i$
- a proof-term: $M : A : \mathsf{Prop}$

$$
\begin{array}{llcll}
\textbf{Terms} & M, N, T, U, A, B & ::= & x \;\mid\; \lambda x : T \,.\, U \;\mid\; M N & \\
& & \mid & \mathsf{Prop} \;\mid\; \mathsf{Type}_i & \\
& & \mid & \Pi x : T \,.\, U & (\mathsf{Type}_i, \mathsf{Type}_j, \mathsf{Type}_{\max(i,j)}) \\
& & \mid & A \Rightarrow B & (\mathsf{Prop}, \mathsf{Prop}, \mathsf{Prop}) \\
& & \mid & \forall x : T \,.\, A & (\mathsf{Type}_i, \mathsf{Prop}, \mathsf{Prop}) \\
& & & & \\
\textbf{Proof-terms} & t, u & ::= & \xi & \\
& & \mid & \lambda \xi^A . \, t \;\mid\; t u & (\mathsf{Prop}, \mathsf{Prop}, \mathsf{Prop}) \\
& & \mid & \lambda x : T \,.\, t \;\mid\; t M & (\mathsf{Type}_i, \mathsf{Prop}, \mathsf{Prop})
\end{array}
$$

$\Rightarrow$ Church's theory of simple types + universes

The intuitionistic logic of $F\omega^2$

(axiom) $$\frac{}{\Gamma \vdash \xi : A}\ {\scriptstyle (\xi : A) \in \Gamma}$$

$(\Rightarrow_I,\ \Rightarrow_E)$ $$\frac{\Gamma; [\xi : A] \vdash t : B}{\Gamma \vdash \lambda \xi^A . t : A \Rightarrow B} \qquad \frac{\Gamma \vdash t : A \Rightarrow B \qquad \Gamma \vdash u : A}{\Gamma \vdash tu : B}$$

$(\forall_I,\ \forall_E)$ $$\frac{\Gamma; [x : T] \vdash t : A}{\Gamma \vdash \lambda x : T . t : \forall x : T . A} \qquad \frac{\Gamma \vdash t : \forall x : T . A \qquad \Gamma \vdash M : T}{\Gamma \vdash tM : A\{x := M\}}$$

$$\begin{array}{lcl}
\bot & \equiv & \forall X : \mathsf{Prop} . X \qquad\qquad \top \ \equiv\ \forall X : \mathsf{Prop} . X \Rightarrow X \\
A \wedge B & \equiv & \forall X : \mathsf{Prop} . (A \Rightarrow B \Rightarrow X) \Rightarrow X \\
A \vee B & \equiv & \forall X : \mathsf{Prop} . (A \Rightarrow X) \Rightarrow (B \Rightarrow X) \Rightarrow X \\
\exists x : T . A(x) & \equiv & \forall X : \mathsf{Prop} . (\forall x : T . A(x) \Rightarrow X) \Rightarrow X \\
M_1 =_T M_2 & \equiv & \forall P : T \to \mathsf{Prop} . P(M_1) \Rightarrow P(M_2)
\end{array}$$

What is Set Theory ?

Introduced by Cantor, Frege, Zermelo, Fraenkel (and many other people) as a framework to formalize potentially all the existing mathematics

$\Rightarrow$ All the fields of classical mathematics can be reduced to Set Theory

Background:	Classical first-order logic with equality
Primitive symbols:	Two binary predicate symbols "$=$" and "$\in$", No primitive constant/function symbols
Formulas:	$\bot, \quad \top, \quad x = y, \quad x \in y,$ $\phi \wedge \psi, \quad \phi \vee \psi, \quad \phi \Rightarrow \psi, \quad \forall x.\phi, \quad \exists x.\phi$
Proofs	Derivations in natural deduction + excluded middle
Axioms	Depends on the theory (Z, ZF, ZFC, etc.)

Zermelo's set theory (1/2)

Equality axioms:	$\forall x.$	$x = x$
	$\forall x, y.$	$x = y \Rightarrow y = x$
	$\forall x, y, z.$	$x = y \Rightarrow y = z \Rightarrow x = z$
Compatibility axioms:	$\forall x, y, z.$	$x = y \Rightarrow y \in z \Rightarrow x \in z$
	$\forall x, y, z.$	$x \in y \Rightarrow y = z \Rightarrow x \in z$
Extensionality axiom:	$\forall x, y$	$(\forall z.\ z \in x \Leftrightarrow z \in y) \Rightarrow x = y$
Other axioms/schemes:	Pairing, Comprehension, Powerset, Union, Infinity	

- We can replace equals by equals in any formula ϕ
  [By induction on ϕ, using equality + compatibility axioms for atomic formulas],
- Extensionality means that sets are determined by their contents only

Zermelo's set theory (2/2)

- Pairing:

  $$\forall a, b \quad \exists c \quad \forall x \qquad x \in c \quad \Leftrightarrow \quad x = a \ \vee \ x = b \qquad\qquad [c \equiv \{a; b\}]$$

- Comprehension:

  $$\forall a \qquad \exists b \quad \forall x \qquad x \in b \quad \Leftrightarrow \quad x \in a \ \wedge \ \phi \qquad\qquad [b \equiv \{x \in a;\ \phi\}]$$

  [for any formula ϕ s.t. $b \notin FV(\phi)$]

- Powerset:

  $$\forall a \qquad \exists b \quad \forall x \qquad x \in b \quad \Leftrightarrow \quad x \subset a \qquad\qquad [b \equiv \mathfrak{P}(a)]$$

- Union:

  $$\forall a \qquad \exists b \quad \forall x \qquad x \in b \quad \Leftrightarrow \quad \exists y \quad y \in a \ \wedge \ x \in y \qquad [b \equiv \bigcup a]$$

- Infinity:

  $$\exists a \qquad 0 \in a \quad \wedge \quad (\forall x \quad x \in a \ \Rightarrow \ s(x) \in a)$$

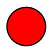
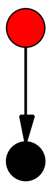
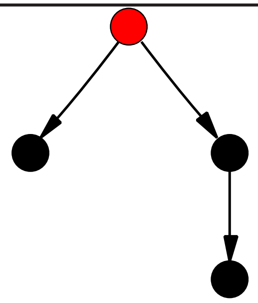
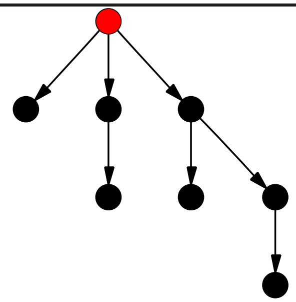
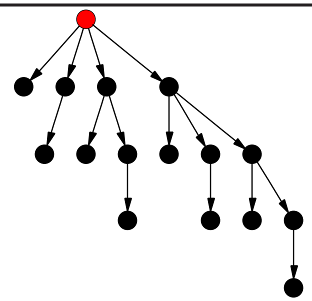
  [where $0 \equiv \varnothing$ and $s(x) \equiv x \cup \{x\}$]

Encoding sets as pointed graphs

Pointed graph $\quad = \quad$ a triple (X, A, a) where

X	:	Type_ℓ	the type of vertices (level ℓ not specified yet)
A	:	$X \to X \to \mathsf{Prop}$	the local membership relation
a	:	X	the root

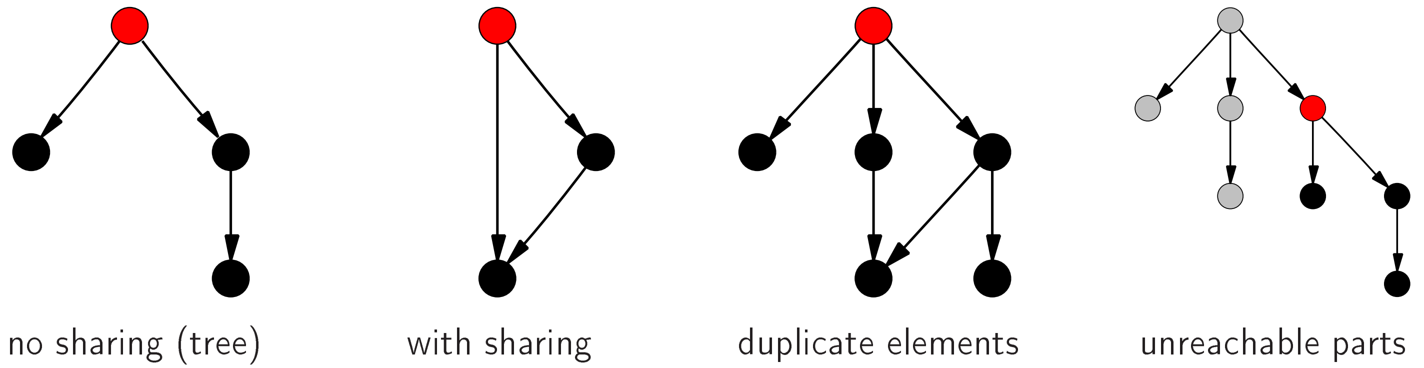
$A(x, y)$ is represented as $\bullet_x \leftarrow \bullet_y$, and the root a as $\bullet_a$

$0 = \varnothing$	$1 = \{0\}$	$2 = \{0; 1\}$	$3 = \{0; 1; 2\}$	$4 = \{0; 1; 2; 3\}$
				

Identifying related pointed graphs

The same set can be represented by several non-isomorphic pointed graphs

Example: the set $2 = \{\varnothing; \{\varnothing\}\}$



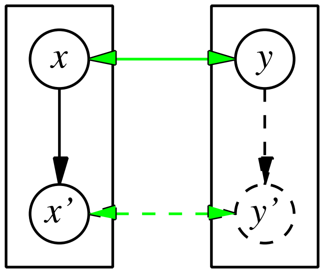
$+$ problems related to (possible) non well-foundedness

Set equality as bisimilarity

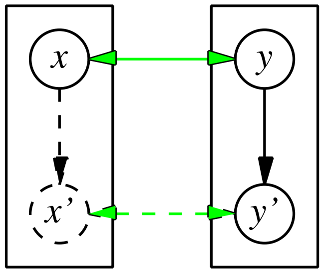
- $R : X \to Y \to \mathsf{Prop}$ is a bisimulation betw. (X, A, a) and (Y, B, b) if:

$$\begin{array}{lllllll}
(1) & & R(a,b) & & & & \\
(2) & \wedge & \forall x, x' : X \,.\, \forall y : Y \,. & A(x', x) \wedge R(x,y) & \Rightarrow & \exists y' : Y \,. & B(y', y) \wedge R(x', y') \\
(3) & \wedge & \forall y, y' : Y \,.\, \forall x : X \,. & B(y', y) \wedge R(x,y) & \Rightarrow & \exists x' : X \,. & A(x', x) \wedge R(x', y')
\end{array}$$

(2)



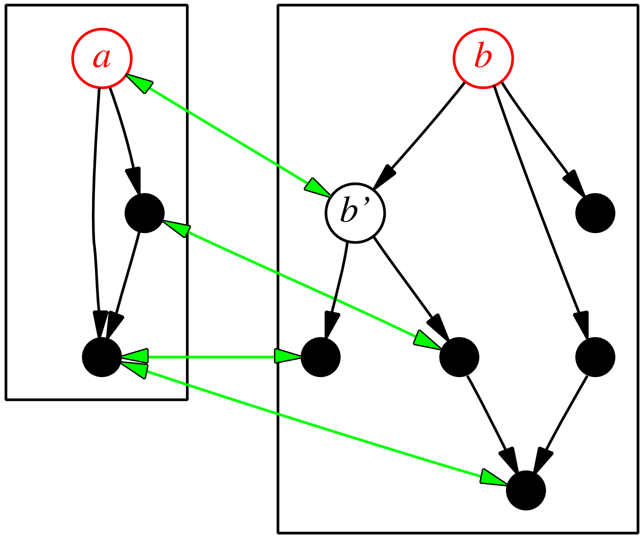
(3)



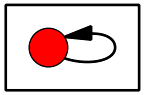
- $(X, A, a) \approx (Y, B, b) \quad \equiv \quad \exists R : X \to Y \to \mathsf{Prop}$ bisimulation(X, A, a, Y, B, b)

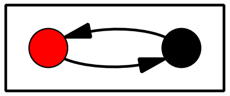
Membership as shifted bisimilarity

$$(X, A, a) \in (Y, B, b) \quad \equiv \quad \exists b' : Y \quad (X, A, a) \approx (Y, B, b') \quad \wedge \quad B(b', b)$$



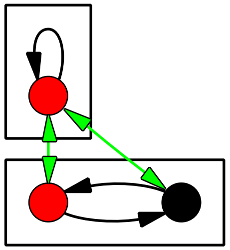
Non well-founded sets

 represents a set x such that $x = \{x\}$

 represents a set y such that $y = \{z\}$ and $z = \{y\}$ for some z

Since there is a bisimulation, we have

$$x = y = z = \{x\} = \{y\} = \{z\}$$



Sets as pointed graphs $+$ equality as a bisimulation

$\Rightarrow$ an immediate interpretation of the Anti-Foundation Axiom (AFA) [P. Aczel]

First properties

- $\approx$ is an equivalence relation

- $\in$ is compatible w.r.t. $\approx$:

$$\begin{array}{lclcl} (X,A,a) \approx (Y,B,b) & \Rightarrow & (Y,B,b) \in (Z,C,c) & \Rightarrow & (X,A,a) \in (Z,C,c) \\ (X,A,a) \in (Y,B,b) & \Rightarrow & (Y,B,b) \approx (Z,C,c) & \Rightarrow & (X,A,a) \in (Z,C,c) \end{array}$$

- Moreover, we have extensionality:

$$\big[\forall (Z,C,c)\,.\quad (Z,C,c) \in (X,A,a) \ \Leftrightarrow\ (Z,C,c) \in (Y,B,b)\big] \quad \Rightarrow \quad (X,A,a) \approx (Y,B,b)$$

- This can be done in many impredicative type systems (CC$_\omega$, ECC, U, U^-, etc.)

Translating formulas

Each variable x_i of set theory is mapped to 3 variables of type theory
$X_i : \mathsf{Type}_\ell$, $A_i : X_i \to X_i \to \mathsf{Prop}$ and $a_i : X_i$.

$$
\begin{array}{lcl}
(x_i = x_j)^\dagger & \equiv & (X_i, A_i, a_i) \approx (X_j, A_j, a_j) \\
(x_i \in x_j)^\dagger & \equiv & (X_i, A_i, a_i) \in (X_j, A_j, a_j) \\
(\bot)^\dagger & \equiv & \bot \\
(\top)^\dagger & \equiv & \top
\end{array}
\qquad
\begin{array}{lcl}
(\phi \wedge \psi)^\dagger & \equiv & \phi^\dagger \wedge \psi^\dagger \\
\cdots & \equiv & \cdots \\
(\forall x_i \,.\, \phi)^\dagger & \equiv & \forall (X_i, A_i, a_i) \,.\, \phi^\dagger \\
(\exists x_i \,.\, \phi)^\dagger & \equiv & \exists (X_i, A_i, a_i) \,.\, \phi^\dagger
\end{array}
$$

Notations:

$$
\begin{array}{lcl}
\forall (X, A, a) \,.\, \phi & \equiv & \forall X : \mathsf{Type}_\ell \,.\, \forall A : (X \to X \to \mathsf{Prop}) \,.\, \forall a : X \,.\; \phi \\
\exists (X, A, a) \,.\, \phi & \equiv & \exists X : \mathsf{Type}_\ell \,.\, \exists A : (X \to X \to \mathsf{Prop}) \,.\, \exists a : X \,.\; \phi \\
\lambda (X, A, a) \,.\, M & \equiv & \lambda X : \mathsf{Type}_\ell \,.\, \lambda A : (X \to X \to \mathsf{Prop}) \,.\, \lambda a : X \,.\; M
\end{array}
$$

A pseudo-sum type

Let X and Y be two types (in Type_ℓ). We want to define a type (in Type_ℓ) that contains disjoint copies of X and Y, plus another element:

$$\begin{array}{lcllcl}
\mathsf{sum}(X,Y) & : & \mathsf{Type}_\ell & := & (X\to\mathsf{Prop}) \to (Y\to\mathsf{Prop}) \to \mathsf{Prop} \\
\mathsf{inl}(X,Y) & : & X \to \mathsf{sum}(X,Y) & := & \lambda x : X \,.\, \lambda f : (X\to\mathsf{Prop}) \,.\, \lambda g : (Y\to\mathsf{Prop}) \,.\, f(x) \\
\mathsf{inr}(X,Y) & : & Y \to \mathsf{sum}(X,Y) & := & \lambda y : Y \,.\, \lambda f : (X\to\mathsf{Prop}) \,.\, \lambda g : (Y\to\mathsf{Prop}) \,.\, g(y) \\
\mathsf{out}(X,Y) & : & \mathsf{sum}(X,Y) & := & \lambda f : (X\to\mathsf{Prop}) \,.\, \lambda g : (Y\to\mathsf{Prop}) \,.\, \bot
\end{array}$$

- $\mathsf{inl}(X,Y)$, $\mathsf{inr}(X,Y)$ and $\mathsf{out}(X,Y)$ behave like constructors:

$$\begin{array}{lllll}
\forall x_1, x_2 : X \,. & \mathsf{inl}(X,Y,x_1) = \mathsf{inl}(X,Y,x_2) & \Rightarrow & x_1 = x_2 \\
\forall y_1, y_2 : Y \,. & \mathsf{inr}(X,Y,y_1) = \mathsf{inr}(X,Y,y_2) & \Rightarrow & y_1 = y_2 \\
\forall x : X \,.\, \forall y : Y \,. & \mathsf{inl}(X,Y,x),\ \mathsf{inr}(X,Y,y),\ \mathsf{out}(X,Y) & \text{pairwise distinct}
\end{array}$$

- No corresponding elimination $\Rightarrow$ existence of "invisible" elements

Translating the pairing axiom

Let (X, A, a) and (Y, B, b) be pointed graphs.

Unordered pair is represented by the pointed graph (T, R, r) given by:

- $T \quad \equiv \quad \mathsf{sum}(X, Y)$
- $r \quad \equiv \quad \mathsf{out}(X, Y)$
- $R(z', z) \quad \equiv$

$$
\begin{array}{ll}
 & \big(\exists x', x : X \;.\; \quad z' = \mathsf{inl}(X, Y, x') \;\;\wedge\;\; z = \mathsf{inl}(X, Y, x) \;\;\wedge\;\; A(x', x)\big) \\
\vee & \big(\exists y', y : Y \;.\; \quad z' = \mathsf{inr}(X, Y, y') \;\;\wedge\;\; z = \mathsf{inr}(X, Y, y) \;\;\wedge\;\; B(y', y)\big) \\
\vee & \big(z' = \mathsf{inl}(X, Y, a) \;\;\wedge\;\; z = \mathsf{out}(X, Y)\big) \\
\vee & \big(z' = \mathsf{inr}(X, Y, b) \;\;\wedge\;\; z = \mathsf{out}(X, Y)\big)
\end{array}
$$

Using this definition, we can prove that:

$$
\forall (Z, C, c) \;.\; \quad (Z, C, c) \in (T, R, r) \quad \Leftrightarrow \quad (Z, C, c) \approx (X, A, a) \;\;\vee\;\; (Z, C, c) \approx (Y, B, b)
$$

Predicates over pointed graphs

In $F\omega^2$, a predicate over pointed graphs is represented by a term

$$P \quad : \quad \Pi X : \mathsf{Type}_\ell \, . \, (X \to X \to \mathsf{Prop}) \to X \to \mathsf{Prop}$$

$\Rightarrow$ Use higher-order quantification to express an infinite number of propositions in $F\omega^2$.

Since we work up to bisimulation, we only consider compatible predicates:

$$\begin{array}{l} \mathsf{COMPAT}(P) \quad \stackrel{\Delta}{\equiv} \\ \qquad \forall (X, A, a), (Y, B, b). \quad (X, A, a) \approx (Y, B, b) \quad \Rightarrow \quad P(X, A, a) \quad \Rightarrow \quad P(Y, B, b) \end{array}$$

Important remark: Any predicate that comes from a formula of set theory

$$P \quad \equiv \quad \lambda(X, A, a) \, . \, \phi^\dagger$$

is compatible (by induction on ϕ).

Translating the comprehension axioms

Let (X, A, a) be a pointed graph, P a compatible predicate over pointed graphs.

The set of elements of (X, A, a) satisfying P is represented by (T, R, r) defined by:

- $T \quad \equiv \quad \mathsf{opt}(X)$
- $r \quad \equiv \quad \mathsf{none}(X)$
- $R(z', z) \quad \equiv$

$$\begin{array}{ll} & (\exists x', x : X \,.\quad z' = \mathsf{some}(X, x') \;\wedge\; z = \mathsf{some}(X, x) \;\wedge\; A(x', x)) \\ \vee & (\exists x' : X \,.\quad z' = \mathsf{some}(X, x') \;\wedge\; z = \mathsf{none}(X) \;\wedge\; A(x', a) \;\wedge\; P(X, A, x')) \end{array}$$

Remark: Definition of option-type $\mathsf{opt}(X) : \mathsf{Type}_\ell$ similar to $\mathsf{sum}(X, Y)$

(but only two constructors: $\mathsf{some}(X) : X \to \mathsf{opt}(X)$ and $\mathsf{none}(X) : \mathsf{opt}(X)$)

Translating the powerset axiom

The powerset of (X, A, a) is represented by the pointed graph (T, R, r) defined by:

- $T \quad \equiv \quad \mathsf{sum}(X, X \to \mathsf{Prop})$
- $r \quad \equiv \quad \mathsf{out}(X)$
- $R(z', z) \quad \equiv$

$$
\begin{array}{lll}
 & \big(\exists x', x : X\ . & z' = \mathsf{inl}(X, X \to \mathsf{Prop}, x') \ \wedge \\
 & & z = \mathsf{inl}(X, X \to \mathsf{Prop}, x) \ \wedge \ A(x', x)\big) \\
\vee & \big(\exists x' : X\ .\ \exists p : X \to \mathsf{Prop}\ . & z' = \mathsf{inl}(X, X \to \mathsf{Prop}, x') \ \wedge \\
 & & z = \mathsf{inr}(X, X \to \mathsf{Prop}, p) \ \wedge \ A(x', a) \ \wedge \ p(x')\big) \\
\vee & \big(\exists p : X \to \mathsf{Prop}\ . & z' = \mathsf{inr}(X, X \to \mathsf{Prop}, p) \ \wedge \ z = \mathsf{out}(X, X \to \mathsf{Prop})\big)
\end{array}
$$

Remark: Each subset $(Y, B, b) \subset (X, A, a)$ is represented by the vertex

$\mathsf{inr}\big(X,\ X \to \mathsf{Prop},\ \lambda x : X\ .\ A(x, a) \wedge (X, A, x) \in (Y, B, b)\big)$ in (T, R, r)

Translating the union axiom

Let (X, A, a) be a pointed graph.

The union of (X, A, a) is represented by the pointed graph (T, R, r) defined by:

- $T \quad \equiv \quad \mathsf{opt}_X$
- $r \quad \equiv \quad \mathsf{none}_X$
- $R(z', z) \quad \equiv$

$$\begin{array}{ll} & \big(\exists x', x : X\ . \quad z' = \mathsf{some}(X, x') \ \wedge\ z = \mathsf{some}(X, x) \ \wedge\ A(x', x)\big) \\ \vee & \big(\exists x', x : X\ . \quad z' = \mathsf{some}(X, x') \ \wedge\ z = \mathsf{none}(X) \ \wedge\ A(x', x) \ \wedge\ A(x, a)\big) \end{array}$$

Predicative Church numerals in Type_2

$$\mathsf{nat} \quad : \quad \mathsf{Type}_2 \quad := \\ \Pi X : \mathsf{Type}_1 \,.\, X \to (X \to X) \to X$$

$$0 \quad : \quad \mathsf{nat} \quad := \\ \lambda X : \mathsf{Type}_1 \,.\, \lambda x : X \,.\, \lambda f : (X \to X) \,.\, x$$

$$\mathsf{S} \quad : \quad \mathsf{nat} \to \mathsf{nat} \quad := \\ \lambda n : \mathsf{nat} \,.\, \lambda X : \mathsf{Type}_1 \,.\, \lambda x : X \,.\, \lambda f : (X \to X) \,.\, f(n(X, x, f))$$

Remark: Normal form of $\overbrace{\mathsf{S}(\cdots \mathsf{S}}^{n \text{ times}}(0) \cdots)$ is Church numeral n:

$$\lambda X : \mathsf{Type}_1 \,.\, \lambda x : X \,.\, \lambda f : (X \to X) \,.\, \underbrace{f(\cdots f}_{n}(x) \cdots)$$

Proposition: $\forall n : \mathsf{nat} \,.\; \neg(\mathsf{S}(n) =_{\mathsf{nat}} 0)$.

Induction principle

Since induction is not provable, define the smallest class $\mathsf{wf\_nat} : \mathsf{nat} \to \mathsf{Prop}$ which contains $\mathsf{0}$ and which is stable by S.

$$\begin{array}{rcl} \mathsf{wf\_nat}(n) & \equiv & \forall P : (\mathsf{nat}{\to}\mathsf{Prop})\,. \\ & & \quad P(0) \quad\Rightarrow\quad \big(\forall p : \mathsf{nat}\,.\, P(p) \Rightarrow P(\mathsf{S}(p))\big) \quad\Rightarrow\quad P(n) \end{array}$$

Proposition:

1. $\mathsf{wf\_nat}(0)$
2. $\forall n : \mathsf{nat}\,.\quad \mathsf{wf\_nat}(n) \Rightarrow \mathsf{wf\_nat}(\mathsf{S}(n))$
3. $\forall P : (\mathsf{nat} \to \mathsf{Prop})\,.\quad P(0) \quad\Rightarrow$
   $\big(\forall n : \mathsf{nat}\,.\ \mathsf{wf\_nat}(n) \ \Rightarrow\ P(n) \ \Rightarrow\ P(\mathsf{S}(n))\big) \quad\Rightarrow$
   $\forall n : \mathsf{nat}\,.\ \mathsf{wf\_nat}(n) \ \Rightarrow\ P(n)$

Injectivity of S

Define a predecessor $\mathsf{pred} : \mathsf{nat} \to \mathsf{nat}$ such that

$$\mathsf{pred}(0) =_\beta 0, \qquad \mathsf{pred}(\mathsf{S}(0)) =_\beta 0, \qquad \mathsf{pred}(\mathsf{S}(\mathsf{S}(n))) =_\beta \mathsf{S}(\mathsf{pred}(\mathsf{S}(n)))$$

(Definition is non trivial, since definition of $\mathsf{nat} : \mathsf{Type}_2$ is predicative.)

Proposition:

1. $\forall n : \mathsf{nat} . \quad \mathsf{wf\_nat}(n) \Rightarrow \mathsf{pred}(\mathsf{S}(n)) =_{\mathsf{nat}} n$
2. $\forall n, m : \mathsf{nat} . \quad \mathsf{wf\_nat}(n) \Rightarrow \mathsf{wf\_nat}(m) \Rightarrow \mathsf{S}(n) =_{\mathsf{nat}} \mathsf{S}(m) \Rightarrow n =_{\mathsf{nat}} m$

Implementing the predecessor function

$$
\begin{array}{llll}
\mathsf{sqr}(X : \mathsf{Type}_1) & : & \mathsf{Type}_1 & := \quad (X \to X \to X) \to X \\[1em]
\mathsf{pair}(X : \mathsf{Type}_1) & : & X \to X \to \mathsf{sqr}(X) & := \quad \lambda x, y : X \,.\, \lambda f : (X \to X \to X) \,.\, f(x, y) \\[1em]
\mathsf{fst}(X : \mathsf{Type}_1) & : & \mathsf{sqr}(X) \to X & := \quad \lambda p : \mathsf{sqr}(X) \,.\, p(\lambda x, y : X \,.\, x) \\
\mathsf{snd}(X : \mathsf{Type}_1) & : & \mathsf{sqr}(X) \to X & := \quad \lambda p : \mathsf{sqr}(X) \,.\, p(\lambda x, y : X \,.\, y)
\end{array}
$$

$$
\begin{aligned}
&\mathsf{step}(X : \mathsf{Type}_1;\ f : X \to X) \quad : \quad \mathsf{sqr}(X) \to \mathsf{sqr}(X) \quad := \\
&\qquad \lambda p : \mathsf{sqr}(X) \,.\, \mathsf{pair}\big(X, \mathsf{snd}(X, p), f(\mathsf{snd}(X, p))\big)
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{pred} \quad : \quad \mathsf{nat} \to \mathsf{nat} \quad := \\
&\qquad \lambda n : \mathsf{nat} \,.\, \lambda X : \mathsf{Type}_1 \,.\, \lambda x : X \,.\, \lambda f : (X \to X) \,. \\
&\qquad\quad \mathsf{fst}\Big(X, n\big(\mathsf{sqr}(X), \mathsf{step}(X, f), \mathsf{pair}(X, x, x)\big)\Big)
\end{aligned}
$$

The crucial step is to remark that: $\mathsf{pred}(\mathsf{S}(\mathsf{S}(n)) \;=_\beta\; \mathsf{S}(\mathsf{pred}(\mathsf{S}(n)))$.

To check that equality, the use of a computer is highly recommended!

Building the set of von Neumann numerals

Idea: the pointed graph $(\mathsf{nat}, <, n)$ already represents von Neumann integer n.

(Strict ordering $n' < n$ is defined using a standard impredicative encoding.)

The set ω is represented by the pointed graph $\big(\mathsf{opt}(\mathsf{nat}), R, \mathsf{none}(\mathsf{nat})\big)$ where

$$
\begin{array}{lll}
R(z, z') \quad \equiv & & \\
& \big(\exists n', n : \mathsf{nat} \,. & \mathsf{wf\_nat}(n') \;\wedge\; \mathsf{wf\_nat}(n) \;\wedge \\
& & z' = \mathsf{some}(\mathsf{nat}, n') \;\wedge\; z = \mathsf{some}(\mathsf{nat}, n) \;\wedge\; n' < n\big) \\
\vee & \big(\exists n' : \mathsf{nat} \,. & \mathsf{wf\_nat}(n') \;\wedge\; z' = \mathsf{some}(\mathsf{nat}, n') \;\wedge\; z = \mathsf{none}(\mathsf{nat})\big)
\end{array}
$$

This pointed graph already satisfies the induction principle (no restriction required)

Intuitionistic Zermelo's set theory

- **We have shown that:**
  - Equality axioms, compatibility axioms and extensionality are valid at any level $\ell \geq 1$
  - Pairing, comprehension, powerset and union are valid at any level $\ell \geq 1$
  - Infinity is valid at level $\ell = 2$ (and, in fact, at any level $\ell \geq 2$)

  $\Rightarrow$ We can derive all Zermelo's axioms at level $\ell = 2$ (in $F\omega.3$)

- Using the intuitionistic natural deduction of $F\omega.3$

  $\Rightarrow$ we can translate any proof of Intuitionistic Zermelo (IZ) in $F\omega.3$:

$$\text{IZ} \quad \leq \quad F\omega.3$$

  [This means that any proof of $\perp$ in IZ induces a proof of $\perp$ in $F\omega.3$]

- **Question:** Can we extend this result to classical Zermelo's set theory ?

Adding Excluded Middle

- Let $\mathbf{cl}$ be a new constant of type $\forall A : \mathsf{Prop}\,.\,\neg\neg A \Rightarrow A$

  $\Rightarrow$ since IZ $\leq F\omega.3$, we have Z $\leq F\omega.3 + \mathbf{cl}$

- To remove the constant $\mathbf{cl}$, we use Coquand-Herbelin's $\neg\neg$-translation:

$$
\begin{array}{lcllcl}
x^+ & \equiv & x & \mathsf{Prop}^+ & \equiv & \mathsf{Prop} \\
(\lambda x : T\,.\,M)^+ & \equiv & \lambda x : T^+\,.\,M^+ & \mathsf{Type}_i^+ & \equiv & \mathsf{Type}_i \\
(MN)^+ & \equiv & M^+N^+ & (\Pi x : T\,.\,U)^+ & \equiv & \Pi x : T^+\,.\,U^+ \\
(M \Rightarrow N)^+ & \equiv & \neg\neg M^+ \Rightarrow \neg\neg N^+ & (\forall x : T\,.\,M)^+ & \equiv & \forall x : T^+\,.\,\neg\neg N^+
\end{array}
$$

  If we set $M^* \equiv \neg\neg M^+$, then we have:

- **Proposition:** *If* A *is provable in* $F\omega.3 + \mathbf{cl}$, *then* A^* *is provable in* $F\omega.3$

- **Theorem:** *If* $F\omega.3$ *is consistent, then (classical) Zermelo is consistent too:*

$$\text{Z} \quad \leq \quad F\omega.3$$

Other results

- With the same number of universes, one gets higher-order for free:
  ho-IZ $\leq$ $F\omega.3$ (Intuitionistic higher-order Zermelo)

- Coquand-Herbelin's $\neg\neg$-translation (in $F\omega.3$) gives excluded middle:
  ho-Z $\leq$ $F\omega.3$ (Classical higher-order Zermelo)

- Predicative universes Type_i $(i \geq 3)$ give Zermelo's universes:
  $\mathsf{Z}^{<\omega\mathsf{u}}$ $\leq$ $F\omega^2$ ($\subset$ CC_ω $\subset$ ECC) (Infinitely many constants for Z-universes)

- **Conjecture:** last inequalities are equivalences (Hint: Melliès-Werner, TYPES'97)

- If we use the same method to encode set theory in systems U and U^-, we get an encoding of Cantor-Frege's (inconsistent) set theory in these systems
  $\Rightarrow$ A new inconsistency proof for systems U and U^- (Russell's paradox)

A PTS with 4 sorts for Zermelo's set theory

- Define $\lambda Z \quad \equiv \quad F\omega.2 \quad + \quad \mathsf{axiom}(\mathsf{Type}_2 : \mathsf{Type}_3) \quad + \quad \mathsf{rule}(\mathsf{Type}_3, \mathsf{Prop}, \mathsf{Prop})$

  (We have: $F\omega.2 \subsetneq \lambda Z \subsetneq F\omega.3$)

- **<u>Claim</u>:** λZ captures the proof-theoretical strength of Zermelo

- Soundness + completeness w.r.t. intuitionistic Zermelo + anti-foundation:

$$\mathsf{IZ} + \mathsf{AFA} \models \phi \quad \Leftrightarrow \quad \lambda Z \models \phi^*$$

  (for any formula ϕ of set-theory)

Extracting programs from proofs

For any proof of IZ, we have a proof-term in $F\omega.3$:

$$\begin{array}{lrcll}
\textbf{Proof-terms} & t, u & ::= & \xi & \text{(axiom)} \\
 & & | & \lambda\xi^A .\, t \quad | \quad tu & (\Rightarrow_I,\ \Rightarrow_E) \\
 & & | & \lambda x : T \,.\, t \quad | \quad tM & (\forall_I,\ \forall_E)
\end{array}$$

Idea: Erase all type annotations/abstractions/applications in proof-terms:

$$\begin{array}{lllllll}
\text{(axiom)} & |\xi| & \equiv & \xi & & & \\
(\Rightarrow_I,\ \Rightarrow_E) & |\lambda\xi^A .\, t| & \equiv & \lambda\xi \,.\, |t| & |tu| & \equiv & |t||u| \\
(\forall_I,\ \forall_E) & |\lambda x : T \,.\, t| & \equiv & |t| & |tM| & \equiv & |t|
\end{array}$$

$\Rightarrow$ Proof-terms become pure λ-terms

Curry-style system $F\omega^2$

- The erased terms are now proof-terms in a new system: Curry-style $F\omega^2$

  $\equiv$ system $F\omega^2$ in which proof-terms are replaced by pure λ-terms

- Semantical twist: $\forall x : T\, .\, \phi$ becomes an intersection (reasonable for impredicativity)

$$\frac{\Gamma; [x : T] \vdash t : A}{\Gamma \vdash t : \forall x : T\, .\, A} \qquad\qquad \frac{\Gamma \vdash t : A \qquad \Gamma \vdash M : T}{\Gamma \vdash t : A\{x := M\}}$$

- If we compose the translation of IZ into $F\omega^2$ with the erasing function, we obtain a non-trivial strongly normalizing Curry-Howard correspondence for Intuitionistic Zermelo
  - types are formulas of set theory
  - proof-terms are pure λ-terms

$\Rightarrow$ Enjoys cut-elimination ($\neq$ Krivine's approach with ZF_ε)

Conclusion

- Strong connections between impredicative PTS and Zermelo's set theory
  - Zermelo's set theory can be encoded in $F\omega.3$
  - Main result: $\mathsf{Z}^{<\omega u} \leq F\omega^2$ $(\subset \mathsf{CC}_\omega \subset \mathsf{ECC})$
  - A PTS (4 sorts) which captures Zermelo's strength
  - The same method can be used to prove the inconsistency of systems U, U^-

- This translation gives a computational contents for proofs of Zermelo

  $\Rightarrow$ Curry-style approach seems to be a better framework

- **Towards a Curry-Howard correspondence in set-theory ?**

  Such a correspondence should be organized along these principles:

  1. $\forall$ is an intersection (i.e. $\neq \Pi$)
  2. cut-elimination ($\neq$ Krivine's approach)