Realizability and forcing

(Curry-Howard correspondence in set theory)

Extend the proof-program correspondence to ZF :

i) We get a type system without the normalization property, but keeping the important thing :
correct computation on data types.
This system enables us to give a precise meaning to the *specification problem*.

ii) We use a generalization of P. Cohen's method in a new setting : *realizability* instead of *forcing*, and *λ-terms* instead of *conditions*.
We get models of ZF *which are not forcing models.*

The main problem : *realize extensionality axiom.*
We define a *conservative extension* ZF_ε of ZF, with three binary symbols : $\in, \subset, \varepsilon$.
$\in$ and $\subset$ have their usual *(extensional)* meaning.
ε is a strong *(intensional)* membership relation.

The theory $\mathrm{ZF}_{\varepsilon}$

Weak *(extensional)* equality: $x = y$ is $x \subset y \wedge y \subset x$.
Strong *(Leibniz)* equality: $x \equiv y$ is $\forall z(x \,\varepsilon\, z \to y \,\varepsilon\, z)$.

0. *Equality & extensionality axioms*

$\forall x \forall y[x \subset y \leftrightarrow (\forall z \,\varepsilon\, x) z \in y]$

$\forall x \forall y[x \in y \leftrightarrow (\exists z \,\varepsilon\, y) x = z]$

1. *Foundation scheme*

$\forall a[(\forall x \,\varepsilon\, a) F(x) \to F(a)] \to \forall a\, F(a)$

2. *Comprehension scheme*

$\forall a \exists b \forall x[x \,\varepsilon\, b \leftrightarrow (x \,\varepsilon\, a \wedge F(x))]$

3. *Pairing axiom*

$\forall a \forall b \exists x[a \,\varepsilon\, x \wedge b \,\varepsilon\, x]$

4. *Union axiom*

$\forall a \exists b(\forall x \,\varepsilon\, a)(\forall y \,\varepsilon\, x)\, y \,\varepsilon\, b$

5. *Power set scheme*

$\forall a \exists b \forall x(\exists y \,\varepsilon\, b) \forall z(z \,\varepsilon\, y \leftrightarrow (z \,\varepsilon\, a \wedge F(z, x)))$

6. *Collection scheme*

$\forall a \exists b(\forall x \,\varepsilon\, a)[\exists y\, F(x, y) \to (\exists y \,\varepsilon\, b) F(x, y)]$

7. *Infinity scheme*

$\forall a \exists b\{a \,\varepsilon\, b \wedge (\forall x \,\varepsilon\, b)[\exists y\, F(x, y) \to (\exists y \,\varepsilon\, b) F(x, y)]\}$

The λ_c-calculus

Λ (resp. Λ_0) is the set of arbitrary (resp. closed) λ-terms. Π is the set of stacks.

1. Any variable x, and the constant cc are λ-terms.
2. If t, u are λ-terms and x is a variable, then tu and $\lambda x\, t$ are λ-terms.
3. If π is a stack, the constant k_π is a λ-term (called a *continuation*).

A *stack* is a sequence $\pi = (t_1, \ldots, t_n, \rho)$, $t_i \in \Lambda_0$, ρ is a *stack constant* (the bottom of the stack) ; $t.\pi$ denotes the stack obtained by pushing t on π.

A *process* is a “ product ” : $t \star \pi$ $(t \in \Lambda_0, \pi \in \Pi)$.

Execution of processes ($\pi, \pi' \in \Pi$) :

$tu \star \pi \succ t \star u.\pi$ $\qquad$ $k_\pi \star t.\pi' \succ t \star \pi$

$\lambda x\, t \star u.\pi \succ t[u/x] \star \pi$ $\qquad$ $\mathrm{cc} \star t.\pi \succ t \star k_\pi.\pi$

$\perp\!\!\!\perp$ is a fixed *cc-saturated* set of processes, i.e. :

$$t \star \pi \in \perp\!\!\!\perp,\ t' \star \pi' \succ t \star \pi \Rightarrow t' \star \pi' \in \perp\!\!\!\perp$$

A *truth value* is a subset of Λ_0 of the form $\mathrm{P} \to \perp\!\!\!\perp$, for any $\mathrm{P} \subset \Pi$.

$$\mathrm{P} \to \perp\!\!\!\perp = \{t \in \Lambda_0\ ;\ (\forall \pi \in \mathrm{P})\, t \star \pi \in \perp\!\!\!\perp\}$$

Realizability

If A is a closed formula of ZF$_\varepsilon$, with parameters :
its truth value is $|A| = \|A\| \to \perp\!\!\!\perp$ with $\|A\| \subset \Pi$;
$t \Vdash A$ is $t \in |A|$ i.e. $(\forall \pi \in \|A\|)\, t \star \pi \in \perp\!\!\!\perp$.

Atomic formulas :
$\|a \not\varepsilon\, b\| = \{\pi \in \Pi\ ;\ (a,\pi) \in b\}$; $\|\bot\| = \Pi$; $\|\top\| = \emptyset$
$\|a \notin b\|$ and $\|a \subset b\|$ are defined later.

Other formulas :
$\|F \to G\| = \{t.\pi\ ;\ t \Vdash F,\ \pi \in \|G\|\}$
$\|\forall x\, F(x)\| = \bigcup_a \|F(a)\|$

A type system for classical logic

Let Γ denote $x_1 : A_1, \ldots, x_n : A_n$ *(a context)*.

1. $\Gamma \vdash x_i : A_i$ $(1 \le i \le n)$
2. $\Gamma \vdash t : A \to B$, $\Gamma \vdash u : A \;\Rightarrow\; \Gamma \vdash tu : B$.
3. $\Gamma, x : A \vdash t : B \;\Rightarrow\; \Gamma \vdash \lambda x\, t : A \to B$.
4. $\Gamma \vdash t : \bot \;\Rightarrow\; \Gamma \vdash t : A$ for every formula A.
5. $\Gamma \vdash t : (A \to B) \to A \;\Rightarrow\; \Gamma \vdash \mathsf{cc}\, t : A$.
6. $\Gamma \vdash t : A \;\Rightarrow\; \Gamma \vdash t : \forall x\, A$ (x is not free in Γ).
7. $\Gamma \vdash t : \forall x\, A \Rightarrow \Gamma \vdash t : A[\tau/x]$ for every term τ.

Realizability is compatible with classical deduction:
Theorem. *If* $x_1 : A_1, \ldots, x_n : A_n \;\vdash\; t : A$ *and if* $t_i \Vdash A_i\,(1 \le i \le n)$*, then* $t[t_1/x_1, \ldots, t_n/x_n] \Vdash A$.

Realization of axioms 1 - 7 of ZF$_\varepsilon$

Foundation scheme

Y $\Vdash \forall a[\forall x(F(x) \to x \notin a) \to \neg F(a)] \to \forall a \neg F(a)$
Y = AA (A = $\lambda a \lambda f\, f.aaf$) is the Turing fixpoint.
Let $t \Vdash \forall a[\forall x(F(x) \to x \notin a) \to \neg F(a)]$, $u \Vdash F(a)$.
We prove Y $\star\, t.u.\pi \in \perp\!\!\!\perp$ by induction on $rk(a)$.
If $v \Vdash F(x)$, $\pi \in \|x \notin a\|$, then $(x, \pi) \in a$, thus $rk(x) < rk(a)$; by induction hypoth., Y$t \star v.\pi \in \perp\!\!\!\perp$.
Thus Y$t \Vdash \forall x(F(x) \to x \notin a)$ and $t \star$ Y$t.u.\pi \in \perp\!\!\!\perp$.
But Y $\star\, t.u.\pi \succ t.$Y$t \star u.\pi \succ t \star$ Y$t.u.\pi \in \perp\!\!\!\perp$.

Comprehension scheme

Given a, let $b = \{(x, t.\pi)\ ;\ t \Vdash F(x),\ (x, \pi) \in a\}$; then $(I, I) \Vdash \forall x[x \notin b \leftrightarrow (F(x) \to x \notin a)]$.
Indeed $\|x \notin b\| = \|F(x) \to x \notin a\|$.

Pairing axiom

If $c = \{a, b\} \times \Pi$, then $I \Vdash a \,\varepsilon\, c$ and $I \Vdash b \,\varepsilon\, c$.
$\|a \notin c\| = \Pi$, thus $I \star t.\pi \succ t \star \pi \in \perp\!\!\!\perp$ if $t \Vdash a \notin c$.

Union axiom

Let $b = Cl(a)$ (transitive closure of a) ; then
$I \Vdash \forall x \forall y[(y \notin x \to x \notin a) \to (y \notin b \to x \notin a)]$.
We prove $\|y \notin b \to x \notin a\| \subset \|y \notin x \to x \notin a\|$:
it is trivial if $\|x \notin a\| = \emptyset$; otherwise $x \in Cl(a)$, thus
$x \subset b$ and $\|y \notin x\| \subset \|y \notin b\|$.

Power set scheme

Let $b = \mathcal{P}(Cl(a) \times \Pi) \times \Pi$. Given x, we find y s.t.
$I \Vdash y \varepsilon b$ and $(I, I) \Vdash \forall z[z \notin y \leftrightarrow (F(z, x) \to z \notin a)]$
Let $y = \{(z, t.\pi) \ ; \ t \Vdash F(z, x), (z, \pi) \in a\}$.
Then $\|y \notin b\| = \Pi$ and $\|z \notin y\| = \|F(z, x) \to z \notin a\|$.

Collection scheme

Given a, we find b such that
$I \Vdash \forall y(F(x, y) \to y \notin b) \to \forall y(F(x, y) \to x \notin a)$.
Let $\Phi(x, p) = \{y$ of minimum rank ; $p \Vdash F(x, y)\}$
and $b = \bigcup\{\Phi(x, p) \times \Pi \ ; \ x \in Cl(a), p \in \Lambda_0\}$.
If $t \Vdash \forall y(F(x, y) \to y \notin b)$, $p \Vdash F(x, y)$, $\pi \in \|x \notin a\|$
then $p \Vdash F(x, y')$ with $y' \in \Phi(x, p)$, $x \in Cl(a)$
(since $(x, \pi) \in a$) ; thus $\|y' \notin b\| = \Pi$ and $t \star p.\pi \in \perp\!\!\!\perp$.

Realization of axioms 0 of ZF$_\varepsilon$

It is the method of *forcing*. We define by induction on $(rk(a) \cup rk(b), rk(a) \cap rk(b))$:

$$\|a \subset b\| = \bigcup_{x \in Cl(a)} \|x \notin b \to x \not\varepsilon a\|$$

$$\|a \notin b\| = \bigcup_{x \in Cl(b)} \|a \subset x, x \subset a \to x \not\varepsilon b\|$$

Then $(I, I) \Vdash a \subset b \leftrightarrow \forall x(x \notin b \to x \not\varepsilon a)$
$(I, I) \Vdash a \notin b \leftrightarrow \forall x(a = x \to x \not\varepsilon b)$.

Example. *If* $\theta = \mathsf{Y}\, \lambda y \lambda x\, xyy$ *then* $\theta \Vdash \forall x(x \subset x)$.

Another method

Let ZF$^-$, written only with ε, be ZF without extensionality and foundation, i.e. axioms 2 - 7 of ZF$_\varepsilon$.

This theory is equiconsistent with ZF (H. Friedman) : we write formulas C(x, y) and E(x, y) such that

ZF$^- \vdash \forall x \forall y[\mathrm{C}(x, y) \leftrightarrow (\forall z\, \varepsilon\, x)\mathrm{E}(z, y)]$
ZF$^- \vdash \forall x \forall y[\mathrm{E}(x, y) \leftrightarrow (\exists z\, \varepsilon\, y)(\mathrm{C}(x, z) \wedge \mathrm{C}(z, x)]$.

Define $x \in y$ by E(x, y) and $x \subset y$ by C(x, y).
Then axioms 0 of ZF$_\varepsilon$ are realized : logical consequences of the already realized axioms of ZF$^-$.

Typed λ-calculus in ZF

Add the function symbols : $\{\}, \cup, \mathcal{P}, \varphi_F, \psi_F, \chi_F$ (for an arbitrary formula F) to the language of ZF$_\varepsilon$, and the following rules to the typed λ_c-calculus :

Foundation scheme.
$\vdash_\varepsilon \mathsf{Y} : \forall x[\forall y(F(y) \to y \notin x) \to \neg F(x)] \to \forall x \neg F(x)$

Pairing axiom. $\vdash_\varepsilon I : x \,\varepsilon\, \{x, y\}$; $\vdash_\varepsilon I : y \,\varepsilon\, \{x, y\}$

Union axiom. $\vdash_\varepsilon I : (z \not\varepsilon\, y \to y \not\varepsilon\, x) \to (z \not\varepsilon \cup x \to y \not\varepsilon\, x)$

Comprehension scheme.
$\vdash_\varepsilon (I, I) : y \not\varepsilon\, \varphi_F(x) \leftrightarrow (F(y) \to y \not\varepsilon\, x)$

Power set scheme. $\vdash_\varepsilon I : \varphi_F(x) \,\varepsilon\, \mathcal{P}(x)$

Collection scheme.
$\vdash_\varepsilon I : \forall z[F(y, z) \to z \not\varepsilon\, \psi_F(x)] \to \forall z[F(y, z) \to y \not\varepsilon\, x]$

Infinity scheme. $\vdash_\varepsilon I : \forall z[F(y, z) \to z \not\varepsilon\, \chi_F(x)] \to$
$\to \forall z[F(y, z) \to y \not\varepsilon\, \chi_F(x)]$; $\vdash_\varepsilon I : x \,\varepsilon\, \chi_F(x)$

Equality & extensionality.
$\vdash_\varepsilon (I, I) : x \notin y \leftrightarrow \forall z(x \subset z, z \subset x \to z \not\varepsilon\, y)$
$\vdash_\varepsilon (I, I) : x \subset y \leftrightarrow \forall z(z \notin y \to z \not\varepsilon\, x)$

Theorem. *If* $\vdash_\varepsilon t : A$ *then* $t \Vdash A$.

More generally, we may add the rule $\vdash_\varepsilon \phi : F$ whenever $\phi \Vdash F$ is provable in ZF.
Let ZF_ε^+ be the set of such formulas.

Normalization

No general normalization theorem (unlike system F).
For example, if $F(x)$ is $x \notin x$ and b is $\varphi_F(a)$, we have by comprehension :

$$\vdash_\varepsilon (I, I) : x \notin b \leftrightarrow (x \notin x \to x \notin a).$$

Thus $\vdash_\varepsilon \delta\delta : b \notin a$ and $\vdash_\varepsilon \lambda x(x)(\delta)\delta : \forall a \exists b(b \notin a)$.

But the important fact is : *the typed λ-calculus in ZF leads to correct computations on data types*.

Example : booleans

$Bool(x)$ is the formula $\forall y(1 \notin y, 0 \notin y \to x \notin y)$ equivalent to $x \equiv 0 \vee x \equiv 1$.

Theorem.
i) If $\vdash_\varepsilon t : Bool(1)$, then t behaves like **true**, *i.e.* $t \star u.v.\pi \succ u \star \pi$ *for closed λ-terms u, v and stack π.*
ii) If $ZF_\varepsilon^+ \vdash Bool(b)$, i.e. $b \equiv 0 \vee b \equiv 1$, for some term b, then $ZF_\varepsilon^+ \vdash b \equiv 0$ or $ZF_\varepsilon^+ \vdash b \equiv 1$.

Realizability models

Λ_c denotes the set of "proof-like" λ-terms, i.e. which contain no constant k_π (no continuation).

Choose $\perp\!\!\!\perp$ such that $|\perp| \cap \Lambda_c = \emptyset$.

The set $\mathfrak{R}$ of truth values is made into a Boolean algebra by means of the preorder :

$$|F| \leq |G| \Leftrightarrow |F \to G| \cap \Lambda_c \neq \emptyset.$$

Any ultrafilter $\mathcal{D}$ on $\mathfrak{R}$ gives a model $\mathcal{M}_\mathcal{D}$ of ZF$_\varepsilon$ and therefore of ZF, such that : $\mathcal{M}_\mathcal{D} \models F \Leftrightarrow |F| \in \mathcal{D}$

Now, even when $\mathcal{D}$ is $\mathcal{M}$-generic, *the model $\mathcal{M}_\mathcal{D}$ is not the forcing model $\mathcal{M}[\mathcal{D}]$*. Indeed :

Some objects in $\mathcal{M}_\mathcal{D}$ have no "name" in $\mathcal{M}$.

There are always non standard integers in $\mathcal{M}_\mathcal{D}$.

It is very likely that $\mathcal{M}_\mathcal{D}$ does not satisfy AC.

The essential difference with the case of forcing lies in the following property : *the intersection of the values of two provable formulas may be false !*

Example: $|\top, \perp \to \perp| \cap |\perp, \top \to \perp| = |\top, \top \to \perp|$

This is due to the lack of *"parallel or"* in λ-calculus.