A basic introduction to reverse mathematics

Hugo Herbelin

6 March 2012

Journées Calculabilité 2012

Paris

Outline

- Reverse mathematics: basics, the big five ($\mathsf{RCA}_0 \subset \mathsf{WKL}_0 \subset \mathsf{ACA}_0 \subset \mathsf{ATR}_0 \subset \Pi^1_1\text{-}\mathsf{CA}_0$)
- Comprehension vs interpolation: towards a uniform presentation of RCA_0, WKL_0, ACA_0, ATR_0, $\Pi^1_1\text{-}\mathsf{CA}_0$
- Towards a computational content to RCA_0, WKL_0, ACA_0, ATR_0, $\Pi^1_1\text{-}\mathsf{CA}_0$

The basis of reverse mathematics

Started by Harvey Friedman in the 70's:

Determine the logical strength underlying standard mathematical theorems, e.g.:

- RCA_0 suffices to prove: Baire category theorem, Intermediate Value Theorem, Soundness of predicate logic, ...
- WKL_0 = Heine/Borel covering lemma = Gödel's Completeness Theorem = Brouwer's Fixed Point Theorem = Separable Hahn/Banach Theorem = Countable commutative rings have a prime ideal = ...
- ACA_0 = Bolzano/Weierstraß Theorem = Countable commutative rings have a maximal ideal = Ramsey's Theorem for colouring's of $[\mathbb{N}]^n$ = ...
- ATR_0 = countable well-orderings are comparable = Perfect Set Theorem = Lusin's separation theorem = ...
- Π^1_1-CA_0 = trees have a largest perfect subtree = Cantor/Bendixson Theorem = Silver's Theorem = ...

A reference book by Stephen Simpson: Subsystems of second-order arithmetic [1999, 2006]

Reverse mathematics: subsystems of second-order arithmetic

- The language of Peano Arithmetic (PA) with its quantification over natural numbers (n, m, ...)
  - Enough to talk about lists, integers, rational numbers, finite sequences, formulae, finite trees, ...
- Quantification over sets of natural numbers (X, Y, ...) and atomic formulae $n \in X$
  - Enough to talk about countable sequences, real numbers, groups, fields, vector spaces, functions over natural numbers, ...
- Induction for those properties definable as a set
- Tuning the strength via comprehension axioms, possibly impacting the strength of induction
  - RCA_0 = Recursive Comprehension Axiom (CA on Δ^0_1-formulae)
  - $\mathsf{WKL}_0 = \mathsf{RCA}_0$ + Weak König's Lemma
  - $\mathsf{ACA}_0 = \mathsf{RCA}_0$ + Arithmetic Comprehension Axiom (CA on Σ^0_1-formulae)
  - $\mathsf{ATR}_0 = \mathsf{ACA}_0$ + Arithmetic Transfinite Recursion
  - Π^1_1-$\mathsf{CA}_0 = \mathsf{ACA}_0$ + Comprehension Axiom on Π^1_1-formulae

Position in Gödel hierarchy

strong	ZF, ZFC, ...		
	Zermelo set theory	System $F_{\omega.2+}$	
	HOL (Church's Simple Type Theory)	Girard's System F_ω	
medium	Z_2 (full 2nd order arithmetic)	Girard-Reynolds' System F	
	Π^1_1-CA_0		$\Psi_0(\Omega_\omega)$
	CZF (Aczel's Constructive Set Theory)		$\Psi_0(\epsilon_{\Omega+1})$ (Bachmann-Howard)
	ATR_0		Γ_0 (Feferman-Schütte)
	ACA_0	Gödel's System T	ϵ_0
	HA^ω (intuit. arithm. in finite types)	Gödel's System T	ϵ_0
	PA	Gödel's System T	ϵ_0
weak	WKL_0	prim. rec. funct.	ω^ω
	RCA_0	prim. rec. funct.	ω^ω
	$I\Sigma_1$	prim. rec. funct.	ω^ω
	PRA (prim. rec. arithmetic)	prim. rec. funct.	ω^ω
	EFA (elementary funct. arithmetic)	prim. rec. funct. up to p^n	ω^3

The arithmetical hierarchy

$$
\begin{array}{lcl}
A \in \Sigma^0_0 & \triangleq & A \text{ is equivalent to some formula } B(n,m,Y) \text{ with } B \text{ primitive recursive} \\
A \in \Pi^0_0 & \triangleq & A \text{ is equivalent to some formula } B(n,m,Y) \text{ with } B \text{ primitive recursive} \\
A \in \Sigma^0_{n+1} & \triangleq & A \text{ is equivalent to some formula } \exists n\, B(n,m,Y) \text{ with } B(n,m,Y) \in \Pi^0_n \\
A \in \Pi^0_{n+1} & \triangleq & A \text{ is equivalent to some formula } \forall n\, B(n,m,Y) \text{ with } B(n,m,Y) \in \Sigma^0_n
\end{array}
$$

$$A \in \Delta^0_n \triangleq A \in \Sigma^0_n \text{ and } A \in \Pi^0_n$$

Formulae in Σ^0_k (or Π^0_k) are said arithmetical

The analytical hierarchy

$$
\begin{array}{lcl}
A \in \Sigma^1_0 & \triangleq & A \text{ is equivalent to some formula } B(m,Y) \in \Delta^0_n \text{ for some } n \\
A \in \Pi^1_0 & \triangleq & A \text{ is equivalent to some formula } B(m,Y) \in \Delta^0_n \text{ for some } n \\
A \in \Sigma^1_{n+1} & \triangleq & A \text{ is equivalent to some formula } \exists X\, B(X,m,Y) \text{ with } B(X,m,Y) \in \Pi^1_n \\
A \in \Pi^1_{n+1} & \triangleq & A \text{ is equivalent to some formula } \forall X\, B(X,,m,Y) \text{ with } B(X,m,Y) \in \Sigma^1_n
\end{array}
$$

$$
A \in \Delta^1_n \quad \triangleq \quad A \in \Sigma^1_n \text{ and } A \in \Pi^1_n
$$

Induction on all sets vs Σ^0_1-induction

Technically, all systems include induction over all sets:

$$\mathsf{IND} : 0 \in X \Rightarrow \forall n(n \in X \Rightarrow n+1 \in X) \Rightarrow \forall n\,(n \in X)$$

but RCA_0 and WKL_0 which have induction over Σ^0_1-formulae:

$$\Sigma^0_1\text{-}\mathsf{IND} : A(0) \Rightarrow \forall n(A(n) \Rightarrow A(n+1)) \Rightarrow \forall n\, A(n)$$

for $A(n) \in \Sigma^0_1$ (possibly with parameters)

However, all 5 systems could be uniformly based on the Σ^0_1-induction scheme: because A can contain free variables n, m, ... of natural numbers and X, Y, ... of sets of natural numbers, the effective strength of Σ^0_1-induction is in fact governed by which sets are definable (i.e. by the comprehension axioms).

Alternatively (AFAIU), all 5 systems could be uniformly based on induction over all sets too, if all prim. rec. functions are taken primitive in RCA_0 and WKL_0.

RCA_0

RCA_0 is characterised by the axiom of Δ^0_1-comprehension:

$$\Delta^0_1\text{-}\mathsf{AC} : \exists X\, \forall n\, (n \in X \iff A(n))$$

for $A(n) \in \Delta^0_1$ (possibly with parameters)

Otherwise said, the only definable sets are the recursive ones.

Because sets definable in RCA_0 are less than Σ^0_1-definable, Δ^0_1-AC does not allow to scale Σ^0_1-induction beyond induction over (closed) Σ^0_1 formulae.

Hence RCA_0 is the second-order variant of $\mathbf{I\Sigma_1}$ which is the restriction of Peano Arithmetic obtained by restricting induction to Σ^0_1-formulae.

The functions provably total in RCA_0 are the primitive recursive functions, hence, its ordinal strength is the one of Primitive Recursive Arithmetic namely ω^ω.

WKL_0

WKL_0 extends RCA_0 with Weak König's Lemma "every infinite binary tree has an infinite path"

$$\mathsf{WKL} : \forall X\, [\text{ tree } X \Rightarrow \forall n\, \exists l\, (|l| > n \wedge\, l \in X) \Rightarrow \exists Y\, \forall n\, \forall l\, (l \approx_n Y \Rightarrow l \in X)]$$

where "tree X" means that $l \in X \Rightarrow l' \in X$ for l' prefix of sequence l and $l \approx_n Y$ means that l is a list of Boolean values reflecting the truth of Y on its n first values (i.e. $\epsilon \approx_0 Y$ and $n \in Y \wedge l \approx_n Y \iff l1 \approx_{n+1} Y$ and $n \notin Y \wedge l \approx_n Y \iff l0 \approx_{n+1} Y$).

Compare to König's lemma: "every infinite *finitely-branching* tree has an infinite path" which is much stronger.

Note: constructive mathematics prefer the contrapositive form, called (the relational form of) Weak Fan Theorem "a binary tree whose branches are bounded is uniformly bounded":

$$\mathsf{WFT} : \forall X\, [\text{ upward closed } X \Rightarrow \forall Y\, \exists n\, \exists l\, (l \approx_n Y \wedge l \in X) \Rightarrow \exists n\, \forall l\, (|l| > n \Rightarrow l \in X)]$$

WKL_0

Sets asserted to exist by **WKL** are in between Δ^0_1-definable and Σ^0_1-definable sets. In particular, **WKL** is not strong enough to make induction exceed the strength of Σ^0_1-induction.

Like for RCA_0, the first-order fragment of WKL_0 is $\mathsf{I\Sigma}_1$.

Like for RCA_0, the functions provably total in WKL_0 are the primitive recursive functions.

Hence, the ordinal strength of WKL_0 is the one of Primitive Recursive Arithmetic, namely ω^ω.

An alternative definition of WKL_0

WKL_0 can be defined without referring to finite sequences: WKL is indeed equivalent over RCA_0 to separation of non-overlapping Σ^0_1-formulae:

$$\Sigma^0_1\text{-}\mathsf{SEP} : \forall n \neg(A_1(n) \wedge A_2(n)) \Rightarrow \exists X \, \forall n \begin{cases} A_1(n) \Rightarrow n \in X \\ A_2(n) \Rightarrow n \notin X \end{cases}$$

for any $A_1(n), A_2(n) \in \Sigma^0_1$

Moreover, Σ^0_1-SEP directly implies Δ^0_1-CA

ACA_0

ACA_0 is characterised by the axiom of Σ^0_1-comprehension:

$$\Sigma^0_1\text{-}\mathsf{AC} : \exists X \, \forall n \, (n \in X \iff A(n))$$

for $A(n) \in \Sigma^0_1$ (possibly with parameters)

Note that, by de Morgan laws, we also have

$$\Pi^0_1\text{-}\mathsf{AC} : \exists X \, \forall n \, (n \in X \iff A(n))$$

for $A(n) \in \Pi^0_1$ (possibly with parameters)

Then, by induction on k, we also obtain Σ^0_k-AC and Π^0_k-AC.

Because any Π^0_k or Σ^0_k-predicate is now definable as a proposition $n \in X$ for some X and hence as a Δ^0_0-formula with set parameters, Σ^1_0-induction now scales to full arithmetical induction.

Henceforth, the first-order part of ACA_0 is Peano Arithmetic. It can prove the termination of all functions of Gödel's System T, and its ordinal strength is ϵ_0.

ATR_0

ATR_0 extends further arithmetical comprehension (i.e. Σ^0_k-CA) using arithmetical transfinite recursion:

$$\mathsf{ATR} : \forall <_J \; (\mathrm{WO}(<_J) \Rightarrow \exists Y \, ((n,j) \in Y \iff j \in J \wedge A(n, Y^j)))$$

where:

- $A(n, X)$ is a Σ^0_1-formula (or any arbitrary arithmetical formula)

- Y^j is the unique set defined by $(n,i) \in Y^j \iff i <_J j \wedge (n,i) \in Y$

- $\mathrm{WO}(<_\mathrm{J})$ means that $<_J$ is a set of pairs defining a well-ordered subset of $\mathbb{N}$

ATR_0

ATR_0 proves the totality of more functions than ACA_0 (a subsystem of System F?)

A result by Friedman, McAloon and Simpson says its ordinal strength is Γ_0.

It is said to be also the ordinal strength of Martin-Löf's type theory with universes but no W-type.

An alternative definition of ATR_0

ATR_0 can be defined without referring to well-orderings: ATR is indeed equivalent to separation of non-overlapping Σ^1_1-formulae:

$$\Sigma^1_1 - \mathsf{SEP} : \forall n \neg (A_1(n) \wedge A_2(n)) \Rightarrow \exists X \, \forall n \begin{cases} A_1(n) \Rightarrow n \in X \\ A_2(n) \Rightarrow n \notin X \end{cases}$$

for any $A_1(n), A_2(n) \in \Sigma^1_1$

Informally speaking:

$$\frac{\mathsf{WKL}_0}{\mathsf{RCA}_0} = \frac{\mathsf{ATR}_0}{\Delta^1_1\text{-}\mathsf{CA}_0}$$

where Δ^1_1-CA_0 extends ACA_0 with Δ^1_1-comprehension, like RCA_0 extended bare second-order arithmetic with Δ^0_1-comprehension

Π^1_1-CA_0

Π^1_1-CA_0 is characterised by the axiom of Σ^1_1-comprehension:

$$\Sigma^1_1\text{-}\mathsf{AC} : \exists X \, \forall n \, (n \in X \iff A(n))$$

for $A(n) \in \Sigma^1_1$ (possibly with parameters)

Note that, by de Morgan laws, we also have

$$\Pi^1_1\text{-}\mathsf{AC} : \exists X \, \forall n \, (n \in X \iff A(n))$$

for $A(n) \in \Pi^1_1$ (possibly with parameters)

However, Σ^1_1-AC and Π^1_1-AC *do not* imply Σ^1_k-AC and Π^1_k-AC (because to be able to nest Σ^1_1-AC one would need the ability to define sets of sets)

Π^1_1-CA$_0$

Π^1_1-CA$_0$ proves the totality of more functions than ATR$_0$ (another subsystem of System F?)

Its ordinal strength is told to be $\Psi_0(\Omega_\omega)$.

We can also generalise the former informal equation:

$$\frac{\mathsf{WKL}_0}{\mathsf{RCA}_0} = \frac{\mathsf{ATR}_0}{\Delta^1_1\text{-}\mathsf{CA}_0} \qquad \text{and} \qquad \frac{\mathsf{WKL}_0}{\mathsf{ACA}_0} = \frac{\mathsf{ATR}_0}{\Pi^1_1\text{-}\mathsf{CA}_0}$$

The big five: summary

system	characterisation	f.o. fragment	functions	ordinal
RCA_0	Δ^0_1-CA	$\mathsf{I}\Sigma_1$	prim. rec.	ω^ω
WKL_0	Σ^0_1-SEP (or WKL)	$\mathsf{I}\Sigma_1$	prim. rec.	ω^ω
ACA_0	Σ^0_1-CA (or Π^0_1-CA)	PA	System T	ϵ_0
ATR_0	Σ^1_1-SEP (or ATR)			Γ_0
Π^1_1-CA_0	Σ^1_1-CA (or Π^1_1-CA)			$\Psi_0(\Omega_\omega)$

The big five: towards a uniform characterisation

(work in progress)

The big five: towards a uniform characterisation

For S_1 and S_2 be classes of formulae, let us call S_1-S_2-interpolation the following scheme:

$$S_1\text{-}S_2\text{-}\mathsf{INTERPOL} : \forall n\,[A_1(n) \Rightarrow A_2(n)] \Rightarrow \exists X\,\forall n \begin{cases} A_1(n) \Rightarrow n \in X \\ n \in X \Rightarrow A_2(n) \end{cases}$$

for $A_1(n) \in S_1$ and $B_2(n) \in S_2$ (possibly with parameters)

We have the following easy facts:

$$\begin{array}{lcl} S\text{-}\mathsf{CA} & \Longleftrightarrow & S\text{-}S\text{-}\mathsf{INTERPOL} \\ S\text{-}\mathsf{SEP} & \Longleftrightarrow & S\text{-}\overline{S}\text{-}\mathsf{INTERPOL} \end{array}$$

where $\overline{S}$ is the complement of S

The big five: revised characterisation

system	characterisation	
RCA_0	Δ^0_1-Δ^0_1-INTERPOL	(i.e. Δ^0_0-CA)
WKL_0	Σ^0_1-Π^0_1-INTERPOL	(i.e. Σ^0_1-SEP)
ACA_0	Σ^0_1-Σ^0_1-INTERPOL or Π^0_1-Π^0_1-INTERPOL	(i.e. Σ^0_1-CA or Π^0_1-CA)
ATR_0	Σ^1_1-Π^1_1-INTERPOL	(i.e. Σ^1_1-SEP)
Π^1_1-CA_0	Σ^1_1-Σ^1_1-INTERPOL or Π^1_1-Π^1_1-INTERPOL	(i.e. Σ^1_1-CA or Π^1_1-CA)

Analysing Δ^0_1-Δ^0_1-INTERPOL

We now show that Δ^0_1-Δ^0_1-**INTERPOL**, i.e. Δ^0_1-**CA** is the same as Π^0_1-Σ^0_1-**INTERPOL**.

($\Rightarrow$) Let $A(n)$, $A_1(n) \in \Pi^0_1$ and $A_2(n) \in \Sigma^0_1$ such that $A(n) \iff A_1(n) \iff A_2(n)$. By taking A_1 and A_n in Π^0_1-Σ^0_1-**INTERPOL**, one gets X such that $A_1(n) \Rightarrow n \in X \Rightarrow A_2(n)$. But then$A_2(n) \Rightarrow A_1(n)$ hence $A(n) \iff n \in X$.

($\Leftarrow$) Let $A_1(n) \iff \forall p\, B_1(n,p) \in \Pi^0_1$ and $A_2(n) \iff \exists p\, B_2(n,p) \in \Sigma^0_1$. If $A_1(n) \Rightarrow A_2(n)$, then we can find a Δ^0_1-formula interpolating $A_1(n)$ and $A_2(n)$. Indeed, taking $A'_1(n) \triangleq \forall p\,(B_1(n,p) \vee \exists p' \leq p\, B_2(n,p'))$ and $A'_2(n) \triangleq \exists p\,(B_2(n,p) \wedge \forall p' < p\, B_1(n,p'))$, we get $A_1(n) \Longrightarrow A'_1(n) \iff A'_2(n) \Longrightarrow A_2(n)$ and we can use Δ^0_1-**CA** with the pair A'_1 and A'_2 to get an interpolant.

The big five: new revised characterisation

system	characterisation
RCA_0	Π^0_1-Σ^0_1-INTERPOL
WKL_0	Σ^0_1-Π^0_1-INTERPOL
ACA_0	Σ^0_1-Σ^0_1-INTERPOL or Π^0_1-Π^0_1-INTERPOL
ATR_0	Σ^1_1-Π^1_1-INTERPOL
Π^1_1-CA_0	Σ^1_1-Σ^1_1-INTERPOL or Π^1_1-Π^1_1-INTERPOL

Towards a computational meaning to the big five

(work in progress)

The "proof-as-program" point of view

Curry [1958]: Hilbert-style propositional logic = simply-typed combinatory logic

Howard [1969]: Gentzen's natural deduction = some simply-typed λ-calculus

Martin-Löf's type theory with W-type [around 1980]: an intuitionistic logic the strength of Π^1_1-CA_0 which is also a programming language

Griffin [1990]: Classical logic = control operator (`callcc`/`throw`)

etc.

We are now going to redefine the "big five" as typed programming languages

Towards a computational meaning to the big five

Comprehension S-**AC** has a standard computational resulting from skolemisation

$$\begin{array}{llcl} \text{formulae} & A & ::= & t \in P \mid t = t \mid A \Rightarrow A \mid A \wedge A \mid A \vee A \\ & & \mid & \forall n\, A \mid \exists n\, A \mid \forall X A \mid \exists X A \\ \text{sets} & P & ::= & X \mid \{n|A\} \end{array}$$

Comprehension rules

$$\frac{\Gamma \vdash A[t/n] \qquad A \in S}{\Gamma \vdash t \in \{n|A\}}\textsc{Compr}_I \qquad \frac{\Gamma \vdash t \in \{n|A\} \qquad A \in S}{\Gamma \vdash A[t/n]}\textsc{Compr}_E$$

Towards a computational meaning to the big five

A similar approach can be taken for S_1-S_2-interpolation:

$$\begin{array}{llll} \text{formulae} & A & ::= & t \in P \mid t = t \mid A \Rightarrow A \mid A \wedge A \mid A \vee A \\ & & \mid & \forall n\, A \mid \exists n\, A \mid \forall X A \mid \exists X A \\ \text{sets} & P & ::= & X \mid \{n | A \triangleright A\} \end{array}$$

Interpolation rules

$$\frac{\Gamma \vdash A_1[t/n] \qquad A_1 \in S_1}{\Gamma \vdash t \in \{n | A_1 \triangleright A_2\}}\,\textsc{Interpol}_I$$

$$\frac{\Gamma \vdash t \in \{n | A_1 \triangleright A_2\} \qquad A_2 \in S_2 \qquad \Gamma, A_1 \vdash A_2}{\Gamma \vdash A_2[t/n]}\,\textsc{Interpol}_E$$

The resulting system (syntax)

$$
\begin{array}{llcl}
\text{formulae} & A, B, C & ::= & t \in P \mid t = t \mid \top \mid \bot \mid A \Rightarrow A \mid A \wedge A \mid A \vee A \\
 & & \mid & \forall n\, A \mid \exists n\, A \mid \forall X\, A \mid \exists X\, A \\
\text{sets} & P & ::= & X \mid \{n|A\} \\
 & & & \\
\text{terms} & t, u & ::= & n \mid 0 \mid t+1 \mid \texttt{rec}\ t\ \texttt{of}\ [t \mid (x,y).t] \\
 & & & \\
\text{proofs} & p, q & ::= & a \mid \iota_i(p) \mid (p_1, p_2) \mid (t, p) \mid (P, p) \mid \lambda a.p \mid \lambda x.p \mid \lambda X.p \mid () \\
 & & \mid & p\, q \mid p\, t \mid p\, P \mid \texttt{absurd}\ p \\
 & & \mid & \texttt{case}\ p\ \texttt{of}\ [a_1.p_1 \mid a_2.p_2] \mid \pi_i\, p \\
 & & \mid & \texttt{dest}\ p\ \texttt{as}\ (x, a)\ \texttt{in}\ q \mid \texttt{dest}\ p\ \texttt{as}\ (X, a)\ \texttt{in}\ q \\
 & & \mid & \texttt{refl} \mid \texttt{subst}\, p\, q \mid \texttt{ind}\ t\ \texttt{of}\ [p \mid (x, a).q] \\
 & & \mid & \texttt{catch}_\alpha p \mid \texttt{throw}_\alpha p \\
 & & \mid & \texttt{compose}\ p\ \texttt{as}\ a\ \texttt{in}\ q
\end{array}
$$

The resulting system (congruence)

The congruence $\equiv$ on formulae is defined as the reflexive transitive compatible closure of the following reduction rules on terms and formulae

$$\begin{array}{lcl}\texttt{rec}\ 0\ \texttt{of}\ [t_0 \,|\, (x,y).t_S] & \rightarrow & t_0 \\ \texttt{rec}\ t+1\ \texttt{of}\ [t_0 \,|\, (x,y).t_S] & \rightarrow & t_S[t/x][\texttt{rec}\ t\ \texttt{of}\ [t_0 \,|\, (x,y).t_S]/y]\end{array}$$

$$\begin{array}{lclcl} 0 & = & t+1 & \rightarrow & \bot \\ t+1 & = & 0 & \rightarrow & \bot \\ 0 & = & 0 & \rightarrow & \top \\ t+1 & = & u+1 & \rightarrow & t = u \end{array}$$

Note: these rules express Peano's axioms, generalising the axioms for $+$ and $\times$ into defining rules for all prim. rec. functions

The resulting system (inference rules, part 1)

$$\frac{(a : A) \in \Gamma}{\Gamma \vdash a : A}\text{AXIOM} \qquad \frac{\Gamma \vdash p : A \qquad A \equiv B}{\Gamma \vdash p : B}\text{CONV}$$

$$\frac{\Gamma \vdash p_1 : A_1 \qquad \Gamma \vdash p_2 : A_2}{\Gamma \vdash (p_1, p_2) : A_1 \wedge A_2}\wedge_I \qquad \frac{\Gamma \vdash p : A_1 \wedge A_2}{\Gamma \vdash \pi_i\, p : A_i}\wedge_E^o$$

$$\frac{\Gamma \vdash p : A_i}{\Gamma \vdash \iota_i(p) : A_1 \vee A_2}\vee_I^i \qquad \frac{\Gamma \vdash p : A_1 \vee A_2 \qquad \Gamma, a_1 : A_1 \vdash p_1 : B \qquad \Gamma, a_2 : A_2 \vdash p_2 : B}{\Gamma \vdash \texttt{case}\ p\ \texttt{of}\ [a_1.p_1 \mid a_2.p_2] : B}\vee_E$$

$$\frac{\Gamma, a : A \vdash p : B}{\Gamma \vdash \lambda a.p : A \Rightarrow B}\Rightarrow_I \qquad \frac{\Gamma \vdash p : A \Rightarrow B \qquad \Gamma \vdash q : A}{\Gamma \vdash p\, q : B}\Rightarrow_E$$

$$\frac{}{\Gamma \vdash () : \top}\top_I \qquad \frac{\Gamma \vdash p : \bot}{\Gamma \vdash \texttt{absurd}\ p : C}\bot_E$$

The resulting system (inference rules, part 2)

$$\frac{\Gamma \vdash p : A(x) \qquad x \text{ fresh}}{\Gamma \vdash \lambda x.p : \forall x\, A(x)}\forall_I \qquad \frac{\Gamma \vdash p : \forall x\, A(x)}{\Gamma \vdash p\, t : A(t)}\forall_E$$

$$\frac{\Gamma \vdash p : A(X) \qquad X \text{ fresh}}{\Gamma \vdash \lambda X.p : \forall X\, A(X)}\forall^2_I \qquad \frac{\Gamma \vdash P : \forall X\, A(X)}{\Gamma \vdash p\, P : A(P)}\forall^2_E$$

$$\frac{\Gamma \vdash p : A(t)}{\Gamma \vdash (t, p) : \exists x\, A(x)}\exists_I \qquad \frac{\Gamma \vdash p : \exists x\, A(x) \qquad \Gamma, a : A(x) \vdash q : B \qquad x \text{ fresh}}{\Gamma \vdash \texttt{dest}\ p\ \texttt{as}\ (x, a)\ \texttt{in}\ q : B}\exists_E$$

$$\frac{\Gamma \vdash p : A(P)}{\Gamma \vdash (P, p) : \exists X\, A(X)}\exists^2_I \qquad \frac{\Gamma \vdash p : \exists X\, A(X) \qquad \Gamma, a : A(X) \vdash q : C \qquad X \text{ fresh}}{\Gamma \vdash \texttt{dest}\ p\ \texttt{as}\ (X, a)\ \texttt{in}\ q : C}\exists^2_E$$

$$\frac{\Gamma, \alpha : T^{\perp\!\!\perp} \vdash p : T}{\Gamma \vdash \texttt{catch}_\alpha\, p : T}\text{CATCH} \qquad \frac{\Gamma \vdash p : T \qquad (\alpha : T^{\perp\!\!\perp}) \in \Gamma}{\Gamma \vdash \texttt{throw}_\alpha\, p : C}\text{THROW}$$

The resulting system (inference rules, part 3)

$$\frac{}{\Gamma \vdash \texttt{refl} : (t = t)}\,\text{REFL} \qquad \frac{\Gamma \vdash p : (t = u) \qquad \Gamma \vdash q : A[t/x]}{\Gamma \vdash \texttt{subst}\ p\ \texttt{in}\ q : A[u/x]}\,\text{SUBST}$$

$$\frac{\Gamma \vdash p : (0 \in P) \qquad \Gamma, a : (n \in P) \vdash q : (n+1 \in P) \qquad n \text{ fresh}}{\Gamma \vdash \texttt{ind}\ t\ \texttt{of}\ [p \,|\, (n,a).q] : (t \in P)}\,\text{IND}$$

$$\frac{\Gamma \vdash p : A_1[t/n] \qquad A_1 \in S_1}{\Gamma \vdash p : (t \in \{n | A_1 \rhd A_2\})}\,\textsc{Interpol}_I$$

$$\frac{\Gamma \vdash p : (t \in \{n | A_1 \rhd A_2\}) \qquad A_2 \in S_2 \qquad \Gamma, a : A_1 \vdash q : A_2 \qquad n \text{ fresh}}{\Gamma \vdash \texttt{compose}\ p\ \texttt{as}\ a\ \texttt{in}\ q : A_2[t/n]}\,\textsc{Interpol}_E$$

The resulting system (notes)

Thanks to `rec`, explicit induction on Σ^0_1-formulae in the case of RCA_0 is no longer needed. Induction can be uniformly formulated on formulae of the form $x \in P$.

The system is fully constructive: it is equipped with a normalisation procedure that we believe to be terminating (by adaptation of the normalisation of System F with interpolation replacing comprehension).

The intuitionistic restriction is immediate to define.

Cross-fertilising computer science and reverse mathematics?

Reverse mathematics described in Simpson's book take place in subsystems of *classical second-order arithmetic* but...

- What is the computational of the equivalence proofs?
- What results become different if intuitionistic logic or type theory is adopted instead?
  - See e.g. Veldman's intuitionistic reverse mathematics program: the functional form of Weak Fan Theorem and Fan Theorem are equivalent!
- What computability can say about reverse mathematics?