Types and Type-Free λ-Calculus

Dana S. Scott, FBA, FNAS

University Professor Emeritus
Carnegie Mellon University
Visiting Scholar
University of California, Berkeley

My Personal Motivation

- ***Denotational semantics*** started in Oxford in late 1969.
- It was hoped that ***domain theory*** would provide a basis both for ***recursive definitions*** in programs and recursive definitions of ***semantical structures***.
- Early troubles were encountered in using ***tops*** and ***bottoms***.
- Soon researchers turned to ***operational semantics***.
- Others wanted to expand work to ***communicating*** and ***parallel processes***.
- ***Axiomatic*** and ***synthetic theories*** did not solve the problems.
- As a result much research effort in domain theory ***faded***.
- Late work by Reynolds and collaborators, however, has opened up ***new*** and ***promising*** approaches.
- Perhaps the much simplified modeling using ***enumeration operators*** can spark new investigations.

Church's λ-Calculus

Definition. λ-calculus — as a formal theory — has rules for the ***explicit definition*** of functions *via* well known equational axioms:

α-conversion

$$\lambda X.[\ldots X\ldots] = \lambda Y.[\ldots Y\ldots]$$

β-conversion

$$(\lambda X.[\ldots X\ldots])(T) = [\ldots T\ldots]$$

η-conversion

$$\lambda X.F(X) = F$$

NOTE: The third axiom will be dropped in favor of a theory employing properties of a partial ordering.

The Enumeration Operator Model

Definitions. (1) ***Pairing:*** $(n,m) = 2^n(2m+1)$.

(2) ***Sequence numbers:*** $\langle\rangle = 0$ and

$\langle n_0, n_1, \ldots, n_{k-1}, n_k\rangle = (\langle n_0, n_1, \ldots, n_{k-1}\rangle, n_k)$.

(3) ***Sets:*** $\mathbf{set}(0) = \varnothing$ and $\mathbf{set}((n,m)) = \mathbf{set}(n) \cup \{m\}$.

(4) ***Kleene star:*** $X^* = \{n \mid \mathbf{set}(n) \subseteq X\}$, for sets $X \subseteq \mathbb{N}$.

Definition. The ***model*** is given by these definitions on ***sets*** of integers:

Application:

$$F(X) = \{ m \mid \exists n \in X^*.(n,m) \in F \}$$

Abstraction:

$$\lambda X.[\ldots X \ldots] =$$

$$\{0\} \cup \{(n,m) \mid m \in [\ldots \mathbf{set}(n) \ldots]\}$$

What is the Secret?

(1) The powerset $\mathcal{P}(\mathbb{N}) = \{ X \mid X \subseteq \mathbb{N} \}$ is a ***topological space*** with the sets $\mathcal{U}_n = \{ X \mid n \in X^* \}$ as a ***basis*** for the topology.

(2) Functions $\Phi : \mathcal{P}(\mathbb{N})^n \longrightarrow \mathcal{P}(\mathbb{N})$ are ***continuous*** iff, for all $m \in \mathbb{N}$, we have $m \in \Phi(X_0, X_1, \ldots, X_{n-1})$ iff there are $k_i \in X_i^*$ for each of the $i<n$, such that $m \in \Phi(\mathbf{set}(k_0), \mathbf{set}(k_1), \ldots, \mathbf{set}(k_{n-1}))$.

(3) The application operation $F(X)$ is continuous as a function of ***two*** variables.

(4) If the function $\Phi(X_0, X_1, \ldots, X_{n-1})$ is continuous, then the abstraction term $\lambda X_0 . \Phi(X_0, X_1, \ldots, X_{n-1})$ is continuous in all of the ***remaining variables***.

(5) If $\Phi(X)$ is continuous, then $\lambda X . \Phi(X)$ is the ***largest set*** F such that for all sets T, we have $F(T) = \Phi(T)$. And, therefore, generally $F \subseteq \lambda X . F(X)$.

NOTE: This model could easily have been defined in 1957!!

It clearly satisfies the rules of α, β-conversion (but not η).

This Lecture is Dedicated to the Memories of

John R. Myhill

Born: 11 August 1923, Birmingham, UK
Died: 15 February 1987, Buffalo, NY

John Shepherdson

Born: 7 June 1926, Huddersfield, UK
Died: 8 January 2015, Bristol, UK

Hartley Rogers, Jr.

Born: 6 July, 1926, Buffalo, NY
Died: 17 July, 2015, Waltham, MA

- John Myhill and John C. Shepherdson, *Effective operations on partial recursive functions*, **Zeitschrift für Mathematische Logik und Grundlagen der Mathematik**, vol. 1 (1955), pp. 310-317.

- Richard M. Friedberg and Hartley Rogers Jr., *Reducibility and completeness for sets of integers,* **Mathematical Logic Quarterly**, vol. 5 (1959), pp. 117-125. Some earlier results are presented in an abstract in **The Journal of Symbolic Logic**, vol. 22 (1957), p. 107.

- Hartley Rogers, Jr., **Theory of Recursive Functions and Effective Computability**, McGraw-Hill, 1967, xix + 482 pp.

Some Lambda Properties

Theorem. For all sets of integers F and G we have:

$$\lambda X.F(X) \subseteq \lambda X.G(X) \text{ iff } \forall X.F(X) \subseteq G(X),$$

$$\lambda X.(F(X) \cap G(X)) = \lambda X.F(X) \cap \lambda X.G(X),$$

and

$$\lambda X.(F(X) \cup G(X)) = \lambda X.F(X) \cup \lambda X.G(X).$$

Definition. A continuous operator $\Phi(X_0, X_1, \ldots, X_{n-1})$ is ***computable*** iff in the model this set is **RE**:

$$F = \lambda X_0 \lambda X_1 \ldots \lambda X_{n-1}.\Phi(X_0, X_1, \ldots, X_{n-1}).$$

How to do Recursion?

Three Basic Theorems.

- All pure λ-terms define ***computable*** operators.
- If $\Phi(X)$ is continuous and if we let $\nabla = \lambda X.\Phi(X(X))$, then the set $P = \nabla(\nabla)$ is the ***least fixed point*** of Φ.
- The least fixed point of a ***computable*** operator is computable.

A Principal Theorem. **These computable operators:**

$$\mathbf{Succ}(X)=\{n+1 \mid n \in X\},$$

$$\mathbf{Pred}(X)=\{n \mid n+1 \in X\},\ \textbf{and}$$

$$\mathbf{Test}(Z)(X)(Y)= \{n \in X \mid 0 \in Z\} \cup \{m \in Y \mid \exists k.k+1 \in Z\},$$

together with λ-calculus, suffice for defining all RE sets.

Gödel Numbering

Theorem. There is a computable $\mathbf{V} = \lambda x.\, \mathbf{V}(x)$ where

(i) $\mathbf{V}(\{0\}) = \lambda y.\lambda x.\, y$,

(ii) $\mathbf{V}(\{1\}) = \lambda z.\lambda y.\lambda x.\, z(x)(y(x))$,

(iii) $\mathbf{V}(\{2\}) =$ **Test**,

(iv) $\mathbf{V}(\{3\}) =$ **Succ**,

(v) $\mathbf{V}(\{4\}) =$ **Pred**, and

(vi) $\mathbf{V}(\{4+(n,m)\}) = \mathbf{V}(\{n\})(\mathbf{V}(\{m\}))$.

Theorem. Every ***recursively enumerable set*** is of the form $\mathbf{V}(\{n\})$.

NOTE: The operator V is the analogue of the Universal Turing Machine.

Inseparable Sets?

Definition. Modify the definition of $\mathbf{V}$ *via* ***finite approximations***:

(i) $\mathbf{V}_k(\{n\}) = \mathbf{V}(\{n\}) \cap \{i \mid i < k\}$ for $n < 5$, and

(ii) $\mathbf{V}_k(\{4+(n,m)\}) = \mathbf{V}_k(\{n\})(\mathbf{V}_k(\{m\}))$.

Theorem. Each $\mathbf{V}_k(\{n\}) \subseteq \mathbf{V}_{k+1}(\{n\})$ is ***finite***,

the predicate $j \in \mathbf{V}_k(\{n\})$ is ***recursive***,

and we have:

$$\mathbf{V}(\{n\}) = \bigcup_{k < \infty} \mathbf{V}_k(\{n\}).$$

Theorem. The sets $\mathcal{L}_0$ and $\mathcal{L}_1$ are ***recursively enumerable***, ***disjoint***, and ***recursively inseparable***:

$$\mathcal{L}_0 = \{n \mid \exists\, j\, [0 \in \mathbf{V}_j(\{n\})(\{n\}) \wedge 1 \notin \mathbf{V}_j(\{n\})(\{n\})]\}$$

$$\mathcal{L}_1 = \{n \mid \exists\, k\, [1 \in \mathbf{V}_k(\{n\})(\{n\}) \wedge 0 \notin \mathbf{V}_k(\{n\})(\{n\})]\}$$

What Happened to the Scott Domains?

Definition. A set $C = \lambda x.C(x)$ represents a ***closure operator*** iff for all $x \subseteq \mathbb{N}$ we have

$$x \subseteq C(x) = C(C(x)).$$

Theorem. The fixed points of closure operators give examples (up to isomorphism) of ***all*** countably based ***algebraic lattices***.

Moreover we have a construction of a ***cartesian closed category*** out of the class of closure operators using these definitions:

$$F\colon C \to D \text{ iff } F = D \circ F \circ C$$

and

$$(C \to D) = \lambda F.\, D \circ F \circ C,$$

where $F \circ G = \lambda x.\, F(G(x))$.

Are There More General Types?

Definition. ***Pairing functions*** for sets in $\mathcal{P}(\mathbb{N})$ can be defined by these ***computable*** enumeration operators:

$\mathbf{Pair}(X)(Y) = (X,Y) = \{2n \mid n \in X\} \cup \{2m+1 \mid m \in Y\}$,

$\mathbf{Fst}(Z) = \{n \mid 2n \in Z\}$, and $\mathbf{Snd}(Z) = \{m \mid 2m+1 \in Z\}$.

Moreover, we may now regard

$\mathcal{P}(\mathbb{N}) = \mathcal{P}(\mathbb{N}) \times \mathcal{P}(\mathbb{N})$, and for $\mathcal{A} \subseteq \mathcal{P}(\mathbb{N})$ we write

$X\ \mathcal{A}\ Y$ iff $(X,Y) \in \mathcal{A}$.

Definition. By a ***type*** over $\mathcal{P}(\mathbb{N})$ we understand a ***(partial) equivalence relation*** $\mathcal{A} \subseteq \mathcal{P}(\mathbb{N})$ where, for all $X,Y,Z \in \mathcal{P}(\mathbb{N})$, we have

$X\ \mathcal{A}\ Y$ implies $Y\ \mathcal{A}\ X$, and

$X\ \mathcal{A}\ Y$ and $Y\ \mathcal{A}\ Z$ imply $X\ \mathcal{A}\ Z$.

Additionally we often write $X{:}\mathcal{A}$ for $X\ \mathcal{A}\ X$.

Let $\mathcal{T}$ be the ***class*** of all such types.

The Category of Types

Definition. The ***exponentiation*** of types $\mathcal{A}, \mathcal{B} \subseteq \mathcal{P}(\mathbb{N})$ is defined as that equivalence relation where

$\mathtt{F}\,(\mathcal{A} \to \mathcal{B})\,\mathtt{G}$ ***iff*** $\forall \mathtt{X}, \mathtt{Y}.\ \mathtt{X}\ \mathcal{A}\ \mathtt{Y}$ implies $\mathtt{F(X)}\ \mathcal{B}\ \mathtt{G(Y)}$.

When we write $\mathtt{F} : \mathcal{A} \to \mathcal{B}$, we should understand the $\mathtt{F}$ as standing for its ***whole*** equivalence class.

Note that $\mathtt{F} : \mathcal{A} \to \mathcal{B}$ ***implies*** $\forall \mathtt{X}.\ \mathtt{X} : \mathcal{A}$ implies $\mathtt{F(X)} : \mathcal{B}$.

Definition. The ***product*** of types $\mathcal{A}, \mathcal{B} \subseteq \mathcal{P}(\mathbb{N})$ is defined as that equivalence relation where

$\mathtt{X}\,(\mathcal{A} \times \mathcal{B})\,\mathtt{Y}$ iff $\mathbf{Fst}\,\mathtt{(X)}\,\mathcal{A}\,\mathbf{Fst}\,\mathtt{(Y)}$ and $\mathbf{Snd}\,\mathtt{(X)}\,\mathcal{B}\,\mathbf{Snd}\,\mathtt{(Y)}$.

And we have

$\mathtt{X} : (\mathcal{A} \times \mathcal{B})$ iff $\mathbf{Fst}\,\mathtt{(X)} : \mathcal{A}$ and $\mathbf{Snd}\,\mathtt{(X)} : \mathcal{B}$.

Note: Types do form a cartesian closed category.

Isomorphic Types

Definition. Two types $\mathcal{A}, \mathcal{B} \subseteq \mathcal{P}(\mathbb{N})$ are ***isomorphic***, in symbols $\mathcal{A} \cong \mathcal{B}$, provided there are sets

$\mathtt{F}:\mathcal{A} \to \mathcal{B}$ and $\mathtt{G}:\mathcal{B} \to \mathcal{A}$ where

$\forall \mathtt{X}:\mathcal{A}.\ \mathtt{X}\ \mathcal{A}\ \mathtt{G(F(X))}$ and $\forall \mathtt{Y}:\mathcal{B}.\ \mathtt{Y}\ \mathcal{B}\ \mathtt{F(G(Y))}$.

Definition. The ***sum*** of types $\mathcal{A}, \mathcal{B} \subseteq \mathcal{P}(\mathbb{N})$ is defined as that equivalence relation where $\mathtt{X}(\mathcal{A}+\mathcal{B})\mathtt{Y}$ iff

either $\exists \mathtt{X}_0, \mathtt{Y}_0 [\, \mathtt{X}_0 \mathcal{A}\, \mathtt{Y}_0 \ \&\ \mathtt{X} = (\{0\}, \mathtt{X}_0) \ \&\ \mathtt{Y} = (\{0\}, \mathtt{Y}_0)\,]$

or $\exists \mathtt{X}_1, \mathtt{Y}_1 [\, \mathtt{X}_1 \mathcal{B}\, \mathtt{Y}_1 \ \&\ \mathtt{X} = (\{1\}, \mathtt{X}_1) \ \&\ \mathtt{Y} = (\{1\}, \mathtt{Y}_1)\,]$.

And we have

$\mathtt{X}:(\mathcal{A}+\mathcal{B})$ iff either $\mathbf{Fst}(\mathtt{X}) = \{0\}$ & $\mathbf{Snd}(\mathtt{X}):\mathcal{A}$

or $\mathbf{Fst}(\mathtt{X}) = \{1\}$ & $\mathbf{Snd}(\mathtt{X}):\mathcal{B}$.

Some Properties of Isomorphism

Theorem. Isomorphism is an ***equivalence relation*** on types,
and if $\mathcal{A}_0 \cong \mathcal{B}_0$ and $\mathcal{A}_1 \cong \mathcal{B}_1$,
then $(\mathcal{A}_0 \times \mathcal{A}_1) \cong (\mathcal{B}_0 \times \mathcal{B}_1)$, $(\mathcal{A}_0 + \mathcal{A}_1) \cong (\mathcal{B}_0 + \mathcal{B}_1)$, and $(\mathcal{A}_0 \to \mathcal{A}_1) \cong (\mathcal{B}_0 \to \mathcal{B}_1)$.

Theorem. We have these ***algebraic laws*** for all $\mathcal{A}, \mathcal{B}, \mathcal{C} \in \mathcal{T}$:

$$(\mathcal{A} \times \mathcal{B}) \cong (\mathcal{B} \times \mathcal{A}) \text{ and } (\mathcal{A} + \mathcal{B}) \cong (\mathcal{B} + \mathcal{A}), \text{ and}$$

$$((\mathcal{A} \times \mathcal{B}) \times \mathcal{C}) \cong (\mathcal{A} \times (\mathcal{B} \times \mathcal{C})),$$

$$((\mathcal{A} + \mathcal{B}) + \mathcal{C}) \cong (\mathcal{A} + (\mathcal{B} + \mathcal{C})),$$

$$(\mathcal{A} \times (\mathcal{B} + \mathcal{C})) \cong (\mathcal{A} \times \mathcal{B}) + (\mathcal{A} \times \mathcal{C}),$$

$$((\mathcal{A} \times \mathcal{B}) \to \mathcal{C}) \cong (\mathcal{A} \to (\mathcal{B} \to \mathcal{C})),$$

$$(\mathcal{A} \to (\mathcal{B} \times \mathcal{C}) \cong (\mathcal{A} \to \mathcal{B}) \times (\mathcal{A} \to \mathcal{C})), \text{ and}$$

$$((\mathcal{A} + \mathcal{B}) \to \mathcal{C}) \cong (\mathcal{A} \to \mathcal{C}) \times (\mathcal{B} \to \mathcal{C})).$$

Types, Subspaces, and Closures

Definition. Every $\mathcal{X} \subseteq \mathcal{P}(\mathbb{N})$ can be considered as a ***topological subspace*** of the the powerset. The ***type*** corresponding to $\mathcal{X}$ is this equivalence relation:

$X [\mathcal{X}] Y$ iff $X = Y \in \mathcal{X}$.

Theorem. The ***topological*** mappings $F : \mathcal{X} \to \mathcal{Y}$ are just the ***restrictions*** of enumeration operators.

Theorem. The ***range*** of a closure operator uniquely determines it.

Definition. The ***type*** corresponding to a closure operator C is this equivalence relation:

$X [C] Y$ iff $X = C(X) = Y$.

Note: The category of **closure operators** can now be considered as a **full subcategory** of the topological category of **subspaces**. The category of subspaces can also be considered as a **full subcategory** of the category of **types**.

Embedding Spaces

Theorem. Every countably based T_0-space $\mathcal{X}$ is homeomorphic to a ***subspace*** of $\mathcal{P}(\mathbb{N})$.

Proof Sketch: Let a subbasis for the topology of $\mathcal{X}$ be $\{ \mathcal{O}_n \mid n \in \mathbb{N} \}$.

Define $\varepsilon : \mathcal{X} \to \mathcal{P}(\mathbb{N})$ by $\varepsilon(x) = \{ n \in \mathbb{N} \mid x \in \mathcal{O}_n \}$.

By the T_0-axiom, this mapping is one-one onto a subspace of $\mathcal{P}(\mathbb{N})$.

Check first that the ***inverse image*** of opens of $\mathcal{P}(\mathbb{N})$ are open in $\mathcal{X}$.

Notice next that $\varepsilon(\mathcal{O}_n) = \varepsilon(\mathcal{X}) \cap \{ S \in \mathcal{P}(\mathbb{N}) \mid n \in S \}$.

Hence, the ***image*** of a open of $\mathcal{X}$ is an open of the subspace.

Therefore, ε is a homeomorphism to a subspace.

Note: This theorem is originally due to:

- P. Alexandroff, *Zur Theorie der topologischen Raume*, C.R. (Doklady) Acad. Sci. URSS, vol. 11 (1936), pp, 55-58.

Dependent Products

Definition. For each $\mathcal{A} \in \mathcal{T}$ the ***identity type*** on $\mathcal{A}$ is defined as that relation such that

$\mathtt{Z}\,(\mathtt{X} \equiv_{\mathcal{A}} \mathtt{Y})\,\mathtt{W}$ iff $\mathtt{Z}\ \mathcal{A}\ \mathtt{X}\ \mathcal{A}\ \mathtt{Y}\ \mathcal{A}\ \mathtt{W}$.

Definition. For each $\mathcal{A} \in \mathcal{T}$, an ***$\mathcal{A}$-indexed family of types*** is a function $\mathcal{B} : \mathcal{P}(\mathbb{N}) \to \mathcal{T}$, such that

$\forall \mathtt{X}_0, \mathtt{X}_1.\ \ \mathtt{X}_0\ \mathcal{A}\ \mathtt{X}_1$ implies $\mathcal{B}(\mathtt{X}_0) = \mathcal{B}(\mathtt{X}_1)$.

Definition. The ***dependent product*** of an $\mathcal{A}$-indexed family of types, $\mathcal{B}$, is this equivalence relation:

$\mathtt{F}_0\,(\prod \mathtt{X} : \mathcal{A}.\,\mathcal{B}(\mathtt{X}))\,\mathtt{F}_1$ iff

$\forall \mathtt{X}_0, \mathtt{X}_1.\ \ \mathtt{X}_0\ \mathcal{A}\ \mathtt{X}_1$ implies $\mathtt{F}_0(\mathtt{X}_0)\ \mathcal{B}(\mathtt{X}_0)\ \mathtt{F}_1(\mathtt{X}_1)$.

Dependent Sums

Definition. The ***dependent sum*** of an $\mathcal{A}$-indexed family of types, $\mathcal{B}$, is this equivalence relation:

$$Z_0 (\sum X{:}\mathcal{A}.\mathcal{B}(X)) Z_1 \text{ iff}$$

$$\exists X_0, Y_0, X_1, Y_1 [\, X_0 \mathcal{A} X_1 \;\&\; Y_0 \mathcal{B}(X_0) Y_1 \;\&$$

$$Z_0 = (X_0, Y_0) \;\&\; Z_1 = (X_1, Y_1) \,]$$

Theorem. The dependent products and dependent sums of indexed families of types are always again types.

Exercise: **Relate dependent sums and products to products and exponents of single types.**

Systems of Dependent Types

Definition. We say that $\mathcal{A},\mathcal{B},\mathcal{C},\mathcal{D}$ form a ***system of dependent types*** iff

- $\forall X_0,X_1.[\,X_0\,\mathcal{A}\,X_1 \Rightarrow \mathcal{B}(X_0)=\mathcal{B}(X_1)\,]$, and
- $\forall X_0,X_1,Y_0,Y_1.[\,X_0\,\mathcal{A}\,X_1\ \&\ Y_0\,\mathcal{B}(X_0)\,Y_1 \Rightarrow \mathcal{C}(X_0,Y_0)=\mathcal{C}(X_1,Y_1)\,]$, and
- $\forall X_0,X_1,Y_0,Y_1,Z_0,Z_1.[\,X_0\,\mathcal{A}\,X_1\ \&\ Y_0\,\mathcal{B}(X_0)\,Y_1\ \&\ Z_0\,\mathcal{C}(X_0,Y_0)\,Z_1 \Rightarrow \mathcal{D}(X_0,Y_0,Z_0)=\mathcal{D}(X_1,Y_1,Z_1)\,]$,

provided that $\mathcal{A}\in\mathcal{T}$, and $\mathcal{B},\mathcal{C},\mathcal{D}$ are functions on $\mathcal{P}(\mathbb{N})$ to $\mathcal{T}$ of the indicated number of arguments.

Theorem. Under the above assumptions on the system $\mathcal{A},\mathcal{B},\mathcal{C},\mathcal{D}$, we will always have

$\prod X{:}\mathcal{A}\,.\sum Y{:}\mathcal{B}(X).\prod Z{:}\mathcal{C}(X,Y).\ \mathcal{D}(X,Y,Z)\in\mathcal{T}$.

Polymorphism Anyone?

Theorem. The class $\mathcal{T}$ of all types is a ***complete lattice***, because it is closed under ***arbitrary intersections***.

Example: $\lambda x.\lambda y.(x,y) : \bigcap_{\mathcal{A},\mathcal{B}} (\mathcal{A} \to (\mathcal{B} \to (\mathcal{A} \times \mathcal{B})))$

Theorem. Any ***monotone*** $\Phi : \mathcal{T} \to \mathcal{T}$ has a least ***fixed point***.

Definition. The ***Scott numerals*** (1963) in the λ-calculus are:

$\underline{0} = \lambda x.\lambda f.x$, $\underline{1} = \lambda x.\lambda f.f(\underline{0})$, $\underline{2} = \lambda x.\lambda f.f(\underline{1})$, etc., and

$\underline{\mathbf{succ}} = \lambda y.\lambda x.\lambda f.f(y)$, and

$\underline{\mathbf{pred}} = \lambda y.y(\underline{0})(\lambda x.x)$.

Example: $\mathcal{S}cott = \bigcap_{\mathcal{A}} (\mathcal{A} \to ((\mathcal{S}cott \to \mathcal{A}) \to \mathcal{A}))$ **types the numerals.**

Propositions as Types?

Definition. Every type $\mathcal{P} \in \mathcal{T}$ can be regarded as a ***proposition***, where ***asserting*** (or ***proving*** $\mathcal{P}$) means finding ***evidence*** $\mathtt{E}:\mathcal{P}$.

Convention: Under this interpretation of logic,
asserting $(\mathcal{P} \times \mathcal{Q})$ means asserting a ***conjunction***,
asserting $(\mathcal{P} + \mathcal{Q})$ means asserting a ***disjunction***,
asserting $(\mathcal{P} \to \mathcal{Q})$ means asserting an ***implication***,
asserting $(\prod \mathtt{X}:\mathcal{A}.\mathcal{P}(\mathtt{X}))$ means asserting a ***universal quantification***, and
asserting $(\sum \mathtt{X}:\mathcal{A}.\mathcal{B}(\mathtt{X}))$ means asserting an ***existential quantification***.

Example: Given $\mathtt{F}:(\mathcal{A} \to (\mathcal{A} \to \mathcal{A}))$**, then asserting**

$$\prod \mathtt{X}:\mathcal{A}.\prod \mathtt{Y}:\mathcal{A}.\prod \mathtt{Z}:\mathcal{A}.\ \mathtt{F}(\mathtt{X})(\mathtt{F}(\mathtt{Y})(\mathtt{Z})) \equiv_{\mathcal{A}} \mathtt{F}(\mathtt{F}(\mathtt{X})(\mathtt{Y}))(\mathtt{Z})$$

is the same as asserting that $\mathtt{F}$ **is an associative binary operation.**

Computability as a Modality?

Definition. For $\mathcal{A} \in \mathcal{T}$ define $\#\mathcal{A} = \mathcal{A} \cap \mathbf{RE}$.

Definition. $[\mathbb{N}]$ is this equivalence relation:

$X[\mathbb{N}]Y$ iff $\exists n \in \mathbb{N}.\ X = \{n\} = Y$.

Theorem. $([\mathbb{N}] \to [\mathbb{N}])$, $([\mathbb{N}] \to ([\mathbb{N}] \cup \{\varnothing\})$,

$\#([\mathbb{N}] \to [\mathbb{N}])$, and $\#([\mathbb{N}] \to ([\mathbb{N}] \cup \{\varnothing\})$

correspond to the ***Baire space***, the superspace of ***partial functions***, the space of **total recursive functions**, and the space of **partial recursive functions**.

Example: For types $\mathcal{A}, \mathcal{B} \in \mathcal{T}$, asserting

$$\# \sum F{:}\mathcal{A} \to \mathcal{B}.\sum G{:}\mathcal{B} \to \mathcal{A}.(\prod X{:}\mathcal{A}.X \equiv_{\mathcal{A}} G(F(X)) \times \prod Y{:}\mathcal{B}.Y \equiv_{\mathcal{B}} F(G(Y)))$$

is the same as asserting that $\mathcal{A}$ and $\mathcal{B}$ are recursively isomorphic.

Toward Descriptive Set Theory?

Wikipedia: Every ***Polish space*** is obtained as a ***continuous image*** of ***Baire space***; in fact, every Polish space is the image of a continuous bijection defined on a closed subset of Baire space.

Similarly, every ***compact*** Polish space is a continuous image of ***Cantor space***.

Because of these ***universality properties***, and because the Baire space $(\mathbb{N} \to \mathbb{N})$ has the convenient property that it is homeomorphic to

$$(\mathbb{N} \to (\mathbb{N} \to \mathbb{N})),$$

many descriptive set-theory results are proved in the context of Baire space alone.

And, a subspace of a Polish space is ***analytic*** iff it is either empty or a continuous image of the Baire space.

Therefore, our category of types is a convenient arena for introducing notions and proofs from this area.

Any Conclusions?

- Enumeration operators over $\mathcal{P}(\mathbb{N})$ are characterized by a simple topology.
- The large category of types over $\mathcal{P}(\mathbb{N})$ inherits much topology.
- λ-calculus over $\mathcal{P}(\mathbb{N})$ plus the arithmetic combinators provides a basic notion of computability.
- The category of types over $\mathcal{P}(\mathbb{N})$ thus inherits aspects of computability.
- Polymorphism for types then provides an abstract foundation for defining inductive and co-inductive data structures.
- Propositions-as-types then will enforce using constructive logic.

The model can in this way function as a laboratory for exploring these ideas in a very concrete fashion.