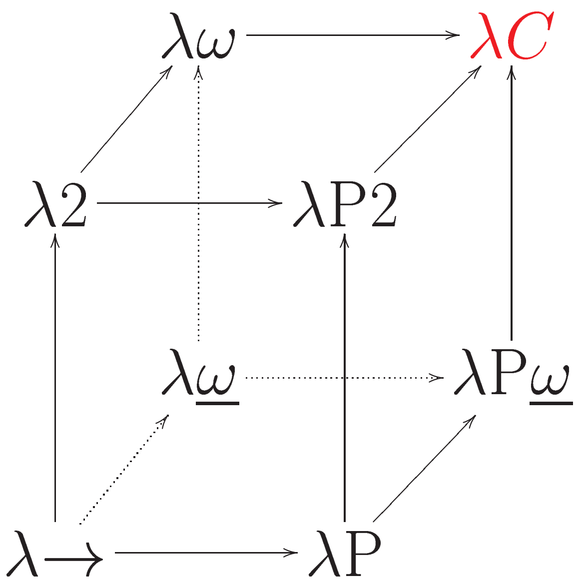
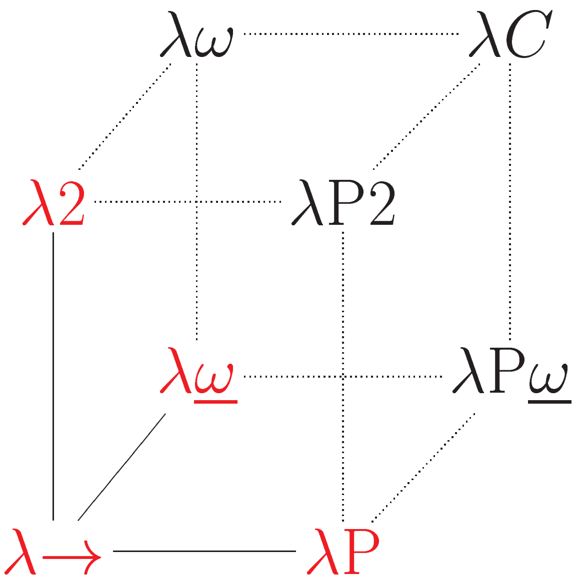
Generation and definability in $\lambda\to$

Henk Barendregt
Nijmegen University

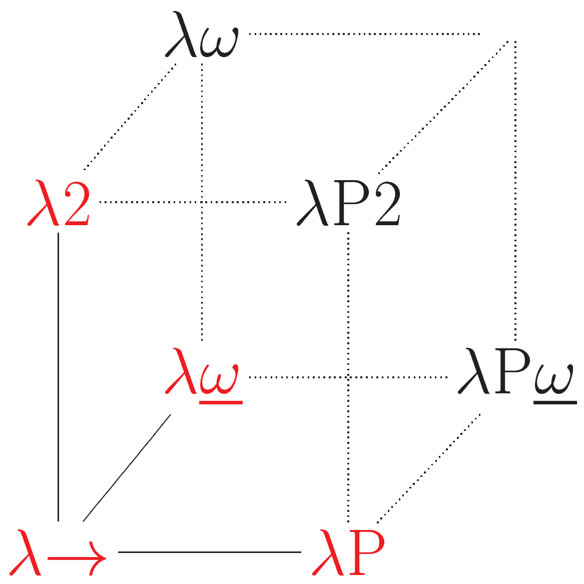
Lambda cube: representing mathematics



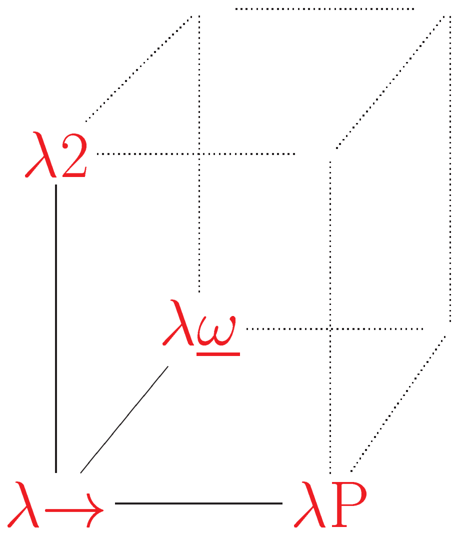
Deconstruction



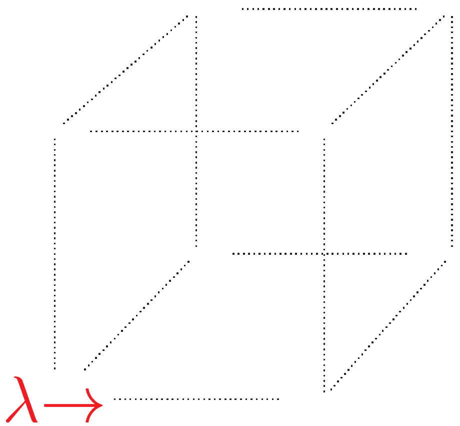
Deconstruction



Deconstruction



Deconstruction



Deconstruction

$\lambda\!\rightarrow$

The system $\lambda{\to}$

Types of $\lambda{\to}$

$$\mathbb{A} ::= \alpha, \beta, \gamma, \ldots$$
$$\mathbb{T} ::= \mathbb{A} \mid \mathbb{T}{\to}\mathbb{T}$$

Terms of type A

$$\boxed{\begin{array}{l} x^A : A \\[1em] \dfrac{M : (A{\to}B) \quad N : A}{(MN) : B} \\[1em] \dfrac{M : B}{(\lambda x^A.M) : (A{\to}B)} \end{array}}$$

Examples $\quad (\alpha{\to}(\alpha{\to}\beta)){\to}(\alpha{\to}\beta) \in \mathsf{T}$

$\vdash (\lambda x^{\alpha\to\alpha\to\beta}\lambda y^{\alpha}.(xy)y) : ((\alpha{\to}(\alpha{\to}\beta)){\to}(\alpha{\to}\beta))$

$\vdash \lambda xy.xyy : (\alpha{\to}\alpha{\to}\beta){\to}(\alpha{\to}\beta)$ (simpler version)

Write

$$\Lambda(A) = \{M \mid \ \vdash M : A\};$$
$$\Lambda^{\emptyset}(A) = \{M \mid \ \vdash M : A \ \& \ \mathrm{FV}(M) = \emptyset\}$$

Conversion

$$\boxed{\begin{array}{lll} (\beta) \quad (\lambda x^A.M)N & = \; M[x := N] & \\ (\eta) \quad \lambda x^A.Mx^A & = \; M & \text{if } x \notin \mathrm{FV}(M) \end{array}}$$

NOTATION. We sometimes leave out superscripts.

$\lambda x^A.x^A \;\rightsquigarrow\; \lambda x^A.x \;\rightsquigarrow\; \lambda x.x.$

FACT. (Curry, de Bruijn & Howard) $\vdash M : A \;\Leftrightarrow\; \vdash_{\mathrm{ML}} A$,

where $\vdash_{\mathrm{ML}}$ denotes probability in minimal propositional logic.

Particular types

$$
\begin{aligned}
0 &:= o \\
n+1 &:= n \to o
\end{aligned}
$$

$$
\begin{aligned}
\mathbb{N} &:= 1 \to 1 &&= (o \to o) \to o \to o \\
\mathsf{T} &:= 1_2 \to 1 &&= (o \to o \to o) \to o \to o \\
\mathsf{M} &:= 3 \to 1 &&= (((o \to o) \to o) \to o) \to o \to o
\end{aligned}
$$

Particular terms

$$
\begin{aligned}
\lambda x^o . x &: 1 \\
\lambda F^2 . F(\lambda x_1^o . F(\lambda x_2^o . F(\lambda x_3^o . x_1))) &: 3 \\
\lambda f^1 x^o . f(f(f\ x)) &: \mathbb{N} \\
\lambda p^{1_2} c^o . pc(tcc) &: \mathsf{T}
\end{aligned}
$$

For these terms only finitely many (bound) variables are needed.

The 'monster' type M (Statman)

$$\mathsf{M} := 3{\to}1 = (((o{\to}o){\to}o){\to}o){\to}o{\to}o$$

Examples of terms of this type

$$\begin{array}{l}
\lambda\Phi^3 c^o.c \\
\lambda\Phi^3 c^o.\Phi(\lambda f^1.fc) \\
\lambda\Phi^3 c^o.\Phi(\lambda f_1^1.\Phi(\lambda f_2^1.f_2(f_1c))) \\
\lambda\Phi^3 c^o.\Phi(\lambda f_1^1.\Phi(\lambda f_2^1.f_2(\Phi(\lambda f_3^1.f_1(f_3c))))) \\
\ldots
\end{array}$$

In general

$$\begin{array}{l}
\lambda\Phi^3 c^o.\Phi(\lambda f_1^1.\vec{w}_1(\Phi(\lambda f_2^1.\vec{w}_2(\ldots\Phi(\lambda f_n^1.\vec{w}_n c))))) = \\
\text{``}\langle\vec{w}_1,\ldots,\vec{w}_n\rangle\text{''},
\end{array}$$

with $\vec{w}_k \in \{f_1,\ldots,f_k\}^*$.

Terms of this type need arbitrarily many (bound) variables. We say that the type M is *rich*. The types $1, 3, \mathbb{N}, \mathsf{T}$ are *poor*.

Representability

Definition. A map $f : \Lambda^{\emptyset}(A) \to \Lambda^{\emptyset}(B)$ is called *representable* iff there exists an $F \in \Lambda^{\emptyset}(A \to B)$ such that for all $M \in \Lambda^{\emptyset}(A)$ one has

$$FM = f(M).$$

Example. The most general elements of $\Lambda^{\emptyset}(\mathbb{N})$ are

$$c_n = \lambda fx.f^{(n)}(x) \text{ (Church's numerals).}$$

The functions $+$ and $\times$ on $\Lambda^{\emptyset}(\mathbb{N})$ defined by

$$\begin{aligned} c_n + c_m &= c_{n+m} \\ c_n \times c_m &= c_{n*m} \end{aligned}$$

are representable:

$$\begin{aligned} F_+ &\equiv \lambda nmfx.nf(mfx) \\ F_\times &\equiv \lambda nmfx.n(mf)x \end{aligned}$$

satisfy

$$\begin{aligned} F_+ c_n c_m &= c_{n+m}; \\ F_\times c_n c_m &= c_{n.m} \end{aligned}$$

Finite generation

DEFINITION. A type A is called *finitely generated* if $\Lambda^{\emptyset}(A)$ can be obtained from finitely many closed terms $\{M_1, \ldots, M_n\}$, not necessarily of type A, by application alone.

EXAMPLES.

(i) $\mathbb{N}$ is finitely generated by $\{c_0, \mathsf{S}^+\}$, where $\mathsf{S}^+ \equiv \lambda nfx.f(nfx)$.

(ii) T is finitely generated by $\{L, P\}$,

where

$$
\begin{aligned}
L &\equiv \lambda b^{1_2} c^{o}.c \\
P &\equiv \lambda t_1 t_2 pc.p(t_1 pc)(t_2 pc)
\end{aligned}
$$

DEFINITION. (Leivant) Let $f : \Lambda^{\emptyset}(A) \to \Lambda^{\emptyset}(B)$

Then f is *slantwise representable* iff

$$\exists F \in \Lambda^{\emptyset}(A^{+} \to B) \forall M \in \Lambda^{\emptyset}(A). f(M) = F M^{+}.$$

Here $+$ is a type substitution operator $+ = [\alpha := A]$ that replaces a base type α by A.

Notice that $\vdash M : A \quad \Rightarrow \vdash M^{+} : A^{+}$.

EXAMPLE. The map exp on $\mathbb{N}$ is slantwise representable. Indeed, $F_e \equiv \lambda n m^+ . m^+ n$, where $+ = [o := o \to o]$, satisfies

$$\begin{aligned} F_e c_n c_m^+ &= \lambda f x . c_m^+ c_n f x \\ &= \lambda f x . (c_n)^{(m)}(f) x \\ &= \lambda f x . f^{(n^m)}(x) \\ &= c_{n^m}. \end{aligned}$$

Enlarged:

$$\begin{aligned} c_2^+ &\equiv (\lambda f^{o\to o} \lambda x^o . f(f(x)))^+ \\ &\equiv \lambda f^{(o\to o)\to o\to o} \lambda x^{o\to o} . f(f(x)) \\ c_2^+ c_3 f x &= c_3(c_3(f))x \\ &= c_3(f)(c_3(f)(c_3(f)(x))) \\ &= f(f(f(f(f(f(f(f(f(x))))))))) \\ &\equiv f^{(9)}(x) \\ &= c_{3^2} f x \end{aligned}$$

Reducibility among types

Definition. Let A, B be types.

(i) (Statman) A is *reducible to* B

$A \leq_1 B \;\Leftrightarrow\;$ there exists a representable injection from $\Lambda^{\emptyset}(A) \to \Lambda^{\emptyset}(B)$.

(ii) (Joly) A *slantwise enumerates* B

$A \geq_s B \;\Leftrightarrow\;$ there is a slantwise representable surjection $\Lambda^{\emptyset}(A) \to \Lambda^{\emptyset}(B)$.

Theorem (Statman). For all types $A \in \mathbb{T}$

$$A \leq_1 \mathsf{T} = (o \to o \to o) \to o \to o.$$

The results of Friedman and Plotkin follow.

Full type structures

Definition. Let for $\alpha \in \mathbb{A}$ a set X_α be given. The *full type structure* over $\vec{X} = \{X_\alpha\}_{\alpha \in \mathbb{A}}$, is the collection of sets

$$\mathcal{M}(\vec{X}) = \{X(A)\}_{A \in \mathbb{T}}$$

defined by

$$\begin{aligned} X(\alpha) &= X_\alpha; \\ X(A \to B) &= X(B)^{X(A)}. \end{aligned}$$

Given a valuation ρ such that $\rho(x^A) \in X(A)$ define $[\![M]\!]_\rho = [\![M]\!]_\rho^{\mathcal{M}(\vec{X})}$ as follows.

$$\begin{aligned} [\![x]\!]_\rho &= \rho(x); \\ [\![MN]\!]_\rho &= [\![M]\!]_\rho [\![N]\!]_\rho; \\ [\![\lambda x^A.M]\!]_\rho &= \boldsymbol{\lambda} d \in X(A).[\![M]\!]_{\rho[x:=d]}. \end{aligned}$$

PROPOSITION. Let $\mathcal{M}(\vec{X})$ be a full type structure.

(i) $M \in \Lambda(A) \;\Rightarrow\; [\![M]\!]_\rho \in X(A)$.

(ii) $M =_{\beta\eta} N \;\Rightarrow\; \mathcal{M}(\vec{X}) \models M = N$.

THEOREM. (H. Friedman) Let $\mathcal{M}(\omega)$ be the full type structure over $\omega = \{0, 1, 2, \ldots\}$. Then

$$M =_{\beta\eta} N \;\Leftrightarrow\; \mathcal{M}(\omega) \models M = N.$$

For finite $\vec{X}$, this never can be the case at types with infinitely many distinct closed terms (e.g. $\mathbb{N}$). But

THEOREM. (Plotkin)

$$M =_{\beta\eta} N \;\Leftrightarrow\; \forall n \in \omega \mathcal{M}(\{0, \ldots, n\}) \models M = N.$$

Definability of elements of a full type structure

Let $\mathcal{M}(\vec{X})$ be a full type structure. An element $d \in X(A)$ is called *definable* iff $d = [\![M]\!]$ for some $M \in \Lambda^{\emptyset}(A)$.

For example in $\mathcal{M}(\{0,1\})$ at type 1 only $\boldsymbol{\lambda} x.x$ is definable.

For $\vec{X}$ finite, the so-called *Plotkin conjecture* stated

$$\{d \in \mathcal{M}(\vec{X}) \mid d \text{ is definable}\} \text{ is decidable.}$$

Loader showed that this is not the case.

Digression **Algebraic geometry**

Let K be a field and let $p \in K[\vec{x}]$. Define

$$V^K(p) = \{\vec{x} \in K^n \mid p(\vec{x}) = 0\}$$

Classical study:

$$p \longmapsto V^{K_0}(p), \text{ for fixed } K_0$$

Digression **Algebraic geometry**

Let K be a field and let $p{\in}K[\vec{x}]$. Define

$$V^K(p) = \{\vec{x}{\in}K^n \mid p(\vec{x}) = 0\}$$

Classical study:

$$p \longmapsto V^{K_0}(p), \text{ for fixed } K_0$$

Modern study (Grothendieck, Dieudonné):

$$K \longmapsto V^K(p_0), \text{ for fixed } p_0$$

Definition. Generalized definability problem.

Let $\vec{X}$ be finite and $A \in \mathbb{T}$. Write $D(\vec{X}, A) = \{d \in \vec{X}(A) \mid d \text{ is definable}\}$.

Problem $\boxed{\bigcup_{\vec{X} \text{ finite}} \bigcup_{A \in \mathbb{T}} D(\vec{X}, A) \text{ decidable?}}$

Loader (1993): No: $\bigcup_{A \in \mathbb{T}} D(\{0, 1, 2, 3, 4, 5, 6\}, A)$ is undecidable.

Joly (2001): No: $\bigcup_{\vec{X} \text{ finite}} D(\vec{X}, \mathsf{M})$ is undecidable.

Write $D(A) = \bigcup_{\vec{X} \text{ finite}} D(\vec{X}, A)$.

THEOREM. (Th. Joly) $D(\mathsf{M})$ is undecidable.

PROOF. Consider register machines M having registers $R_1, \ldots, R_m$, with content $r_1, \ldots, r_m$ respectively, as states $Q = \{s, q_1, \ldots, q_p, h\}$ and as statements

INC(k, q_i)	$r_k := r_k + 1$; goto q_i
JZD(k, q_i, q_j)	if $r_k = 0$ goto q_i else $[r_k := r_k \dot{-} 1$; goto $q_j]$.

The halting problem asks whether machine M with empty input halts. We can arrange it so that the machine always will halt with the empty state if it halts at all.

Moral: One has for

M	a register machine
M	a lambda term representing a computation of M
M	the monster type
$\mathcal{M}$	the finite model above

the potential infinity (concerning both space and time) of the computation of M, *not available in any finite type structure, can be captured by a term* M *of type* M *of which the correctness can be checked inside the model* $\mathcal{M}$.

Intuitively, there are

terms C corresponding to each configuration $c = (q, r_1, \ldots, r_m)$;

if $c_0, c_1, \ldots, c_n$ is a terminating computation of M, then

$$M = \Sigma_{i=0}^{n} C_i$$

(for suitable Σ)

An example of a terminating machine with two registers

Machine	Configuration c_i	Encoding term $M = \Sigma.c_i$ with $\lambda\Phi c.M \in \Lambda(\mathsf{M})$	
$s:\ \mathrm{INC}(1,1)$	$(s,0,0)$	c	
$1:\ \mathrm{INC}(1,2)$	$(1,1,0)$	$f_1 c$	
$2:\ \mathrm{INC}(2,3)$	$(2,2,0)$	$f_1 f_1 c$	$\equiv f_1(f_1 c)$
$3:\ \mathrm{JZD}(2,3,4)$	$(3,2,1)$	$f_2 f_1 f_1 c$	
$4:\ \mathrm{JZD}(1,h,4)$	$(4,2,0)$	$\Phi\lambda z.z f_1 f_1 c$	$\equiv \Phi(\lambda z.z f_1 f_1 c)$
	$(4,1,0)$	$\Phi\lambda z_1.\Phi\lambda z.z z_1 f_1 c$	
	$(4,0,0)$	$\Phi\lambda z_2.\Phi\lambda z_1.\Phi\lambda z.z z_1 z_2 c$	
	$(h,0,0)$		

Taking as $X_0 = \{s,1,2,3,4,h,\top,\bot\} \times \{0,1\}^2$ the correctness of computations can be verified in the model $\mathcal{M}(\vec{X})$ from the image of these terms under a valuation (giving the right interpretation for Φ and c). Here elements of $\{0,1\}^2$ correspond to the abstract interpretation $\langle \mathrm{sg}(r_1), \mathrm{sg}(r_2)\rangle$ and $\top,\bot$ are auxiliary 'states'.

$$
\begin{array}{lcl}
\text{M halts} & \Leftrightarrow & \text{there is a terminating computation } C \\
 & \Leftrightarrow ! & \exists P \in \Lambda^{\Phi:3,c:o}(o).[\![P]\!]_\rho = (h, \langle 0,0\rangle) \\
 & \Leftrightarrow & \exists P \in \Lambda^{\Phi:3,c:o}(o).[\![\lambda\Phi c.P]\!]\,\rho(\Phi)\,\rho(c) = (h, \langle 0,0\rangle) \\
 & \Leftrightarrow & \exists d \in D(\vec{X}, \mathsf{M}).d\,\rho(\Phi)\,\rho(c) = (h, \langle 0,0\rangle)
\end{array}
$$

This shows that $D(\vec{X}, \mathsf{M})$ cannot be decidable (uniformly in $\vec{X}$). ■

Detail 1. **Regular languages and type structures**

Given a finite automaton M$= (Q, \Sigma, q_0, \delta, F)$, let $L^r(\mathrm{M})$ be the language of words in Σ^* accepted by this machine reading right to left. Let $\Sigma = \{a, b, c\}$, say. Define $W = 1^3{\to}o{\to}o$. For e.g.

$$w = abaca,$$

define

$$\underline{w} = \lambda a^1 b^1 c^1 x^o.abacax,$$

with parentheses to the right. Then in the type structure over Q

$$w{\in}L^r(\mathrm{M}) \ \Leftrightarrow\ [\![\underline{w}]\!]^{\mathcal{M}(Q)}\delta_a\delta_b\delta_c q_0{\in}F,$$

where $\delta_a = \boldsymbol{\lambda} q.\delta(a, q) : Q(1)$ etcetera.

Detail 2. **Recognizing λI-terms**

Define the following language

$$t ::= c \mid f^1 t \mid \Phi^3 \lambda f^1 t$$

Then for any t in this language one has $\lambda\Phi c.t \in \mathsf{M}$. We want to represent in a model $\mathcal{M}$ the notion $t \in \lambda\mathsf{I}$.

$$\Phi\lambda f \Phi\lambda g.f g g c$$

The idea is to interpret Φ such that it takes its argument

$$F \equiv \lambda f. \dots$$

and applies it to TWO functions ($\mathsf{K}\bot$ and I) in order to determine $[\![\Phi F]\!]_\rho$. This is necessary since

$$\lambda f.t \in \lambda\mathsf{I} \;\Leftrightarrow\; f \in \mathrm{FV}(t) \;\&\; t \in \lambda\mathsf{I}.$$

THEOREM. (Th. Joly) Let $A \in \mathbb{T}$. Then the following are equivalent.

(i) A is finitely generated.

(ii) $D(A)$ is decidable.

(iii) $A \not\geq_s \mathsf{M}$.

(iv) A is poor. %%%($\Leftrightarrow$ $\mathsf{T} \geq_s A$).

PROOF. We will sketch the proofs of

$$(i) \Rightarrow (ii) \Rightarrow (iii) \Rightarrow (iv) \Rightarrow (i).$$

$(i) \Rightarrow (ii)$. Suppose $\Lambda^{\emptyset}(A)$ is generated by $\{M_1, \ldots, M_p\}$, with $\vdash M_i : B_i$. Let $d \in \vec{X}(A)$ in order to determine whether d is definable. The elements of $\Lambda^{\emptyset}(A)$ are applications of the $\vec{M}$, so can only involve combinations having as type one of the subtypes of the $\vec{B}$. Let these types be $C_1, \ldots, C_q$. Since $\vec{X}(A) \subseteq \bigcup_{C_j} \vec{X}(C_j)$ is finite, we just wait to see if d appears as an $[\![N]\!]$, where N is such an applicative combination. The process stops when new combinations do not give new elements.

$(ii) \Rightarrow (iii)$. We first notice that

$$A \geq_s B \;\&\; D(A) \text{ decidable } \Rightarrow D(B) \text{ decidable.} \tag{1}$$

Indeed, suppose that $D(A)$ is decidable and $A \geq_s B$. If the surjection $f : \Lambda^{\emptyset}(A) \rightarrow \Lambda^{\emptyset}(B)$ is not slantwise but directly representable by F, then

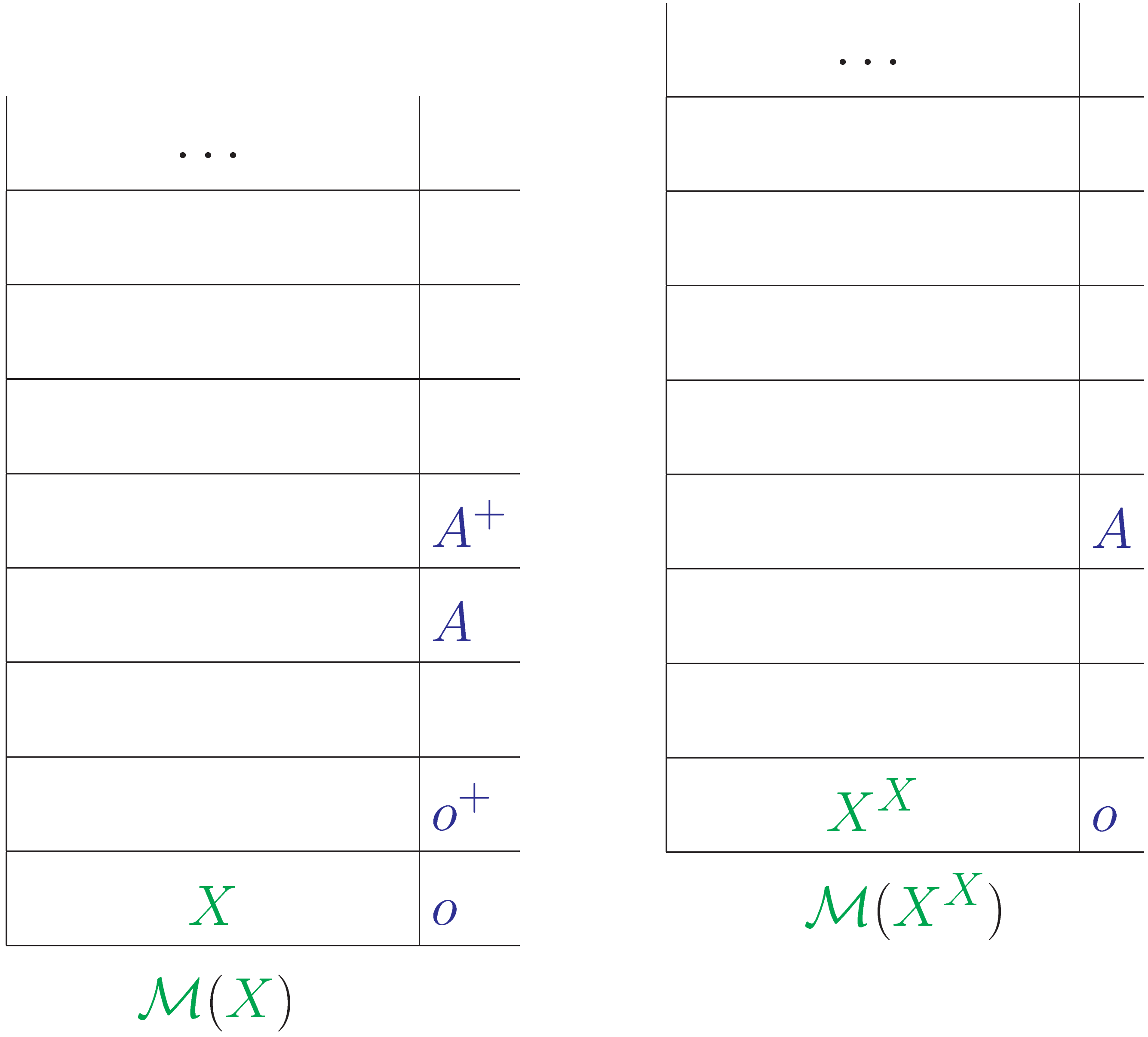
$$d \in D(\vec{X}, B) \Leftrightarrow d \in [\![F]\!](D(\vec{X}, A)),$$

and decidability of $D(B)$ follows. If, however, $f(M) = FM^{+}$, then

$$\begin{aligned} d \in D(\vec{X}, B) &\Leftrightarrow d \in [\![F]\!](D(\vec{X}, A^{+})) \\ &\Leftrightarrow d \in [\![F]\!](D(\vec{X}^{+}, A)), \end{aligned}$$

with $X^{+}_{\alpha} = \vec{X}(\alpha^{+})$, so $\vec{X}(A^{+}) = \vec{X}^{+}(A)$ and decidability follows again!

Now by (1) and the undecidability of $D(\mathsf{M})$, it follows that $A \not\geq_s \mathsf{M}$ for A with decidable $D(A)$.



$$+ = [o := o \to o]$$

$(iii) \Rightarrow (iv)$. This follows from a proposition that we will not prove:

$$A \text{ is rich} \quad \Rightarrow \quad A \geq_s \mathsf{M}.$$

Steps in the proof are

$$\begin{aligned} A &\geq_s \mathbb{L}_A \text{ for all types} \\ \mathbb{L}_A &\geq_s \mathsf{W} = (((\alpha{\to}\alpha){\to}o){\to}o){\to}(\alpha{\to}o){\to}\alpha{\to}o \text{ if } A \text{ is rich} \\ \mathsf{W} &\geq_s \mathsf{M} \end{aligned}$$

$(iv) \Rightarrow (i)$.

$x{:}A \vdash^{I} x : A$	$\lambda x.x$
$\dfrac{\Gamma \vdash^{I} M : A \to B \quad \Gamma \vdash^{I} N : A}{\Gamma \vdash^{I} MN : B}$	$\mathsf{S}_{\Gamma, A\to B} = \lambda ab\vec{x}.a\vec{x}(b\vec{x})$
$\dfrac{\Gamma, y{:}A \vdash^{I} M : B}{\Gamma \vdash^{I} \lambda y.M : A \to B}$	$\mathsf{I}_{\Gamma, y:A, A\to B} = \lambda a\vec{x}y.a\vec{x}y$
$\dfrac{\Gamma \vdash^{I} M : B}{\Gamma \vdash^{I} \lambda y.M : A \to B}$ y is fresh	$\mathsf{K}_{\Gamma, A\to B} = \lambda a\vec{x}y.a\vec{x}$

Then $\qquad \Gamma \vdash M : A \;\Leftrightarrow\; \Gamma \upharpoonright \mathrm{FV}(M) \vdash^{I} M : A$

$$\vec{x}{:}\vec{A} \vdash^{I} M : A \;\&\; M \text{ in nf} \;\Rightarrow\; \lambda\vec{x}.M \in \mathcal{C}(\vec{x}, \mathrm{Var}(M)),$$
$$\text{some FG class } \mathcal{C}(\vec{x}, \mathrm{Var}(M))$$

Hence in case A is poor one has $\vdash M : A \;\Rightarrow\; M \in \mathcal{C}$, $\mathcal{C}$ FG.

Thus $\Lambda^{\emptyset}(A)$ is FG.

As a corollary to the undecidability of $D(\mathsf{M})$ Joly has obtained the recent result of Loader that β-matching

$$\exists \vec{X}. M \vec{X} =_\beta N$$

is undecidable. Still open is whether $\beta\eta$-matching is decidable.