# Reflections on a Theorem of Henkin

**William Gunther and Richard Statman**

**Abstract** The $\lambda\delta$-calculus is the $\lambda$-calculus augmented with a discriminator which distinguishes terms. We consider the simply typed $\lambda\delta$-calculus over one atomic type variable augmented additionally with an existential quantifier and a description operator, all of lowest type. First we provide a proof of a folklore result which states that a function in the full type structure of $[n]$ is $\lambda\delta$-definable from the description operator and existential quantifier if and only if it is symmetric, that is, fixed under the group action of the symmetric group of $n$ elements. This proof uses only elementary facts from algebra and a way to reduce arbitrary functions to functions of lowest type via a theorem of Henkin. Then we prove a necessary and sufficient condition for a function on $[n]$ to be $\lambda\delta$-definable without the description operator or existential quantifier, which requires a stronger notion of symmetry.

**Keywords** Lambda calculus · Lambda delta calculus · Types · Typed lambda calculus · Simply typed lambda calculus · Type theory · Classical type theory · First-order logic · Henkin semantics · Typed lambda calculus semantics · Delta discriminator · Description operator

## 1 Introduction

Let $\mathscr{M}^n$ be the full type structure over a ground domain of size $n$. It is folklore that a member of $\mathscr{M}^n$ is symmetric if and only if it is definable in type theory. The origin of this theorem is murky. It is safe to say that it was not known to Newton. Robin Gandy told the senior author that he knew it in the 1940s. It is not unlikely it was known to Church before this. It occurred to the senior author as a student in the 1970s after reading Andrews [1] and Lauchli [5]. There are not many proofs in the literature. A proof appears in Van Benthem [8] in the 1990s, but it is incomplete. A proof follows immediately from an observation due to Leon Henkin [4].

In this note, we intend to do two things. First, we shall generalize the folklore theorem to Church's $\lambda\delta$-calculus [3] for the $\mathscr{M}^n$, and also to the "profinite" model which is the "limit" of the $\mathscr{M}^n$. Second, we shall provide a straightforward proof of the folklore theorem itself using only simple facts about the symmetric group and its action on equivalence relations.

 
M. Manzano et al. (eds.), *The Life and Work of Leon Henkin*, Studies in Universal Logic,
DOI 10.1007/978-3-319-09719-0_14

# 2 Preliminaries

We first do a review of the lambda calculus, simple types, Henkin-style semantics, and extending simple typed lambda calculus to type theory.

## 2.1 Lambda Calculus

### Untyped Lambda Calculus

The untyped lambda calculus serves as the underlying language of our more disciplined systems. We give a quick reminder.

**Definition 1** Fix some countable set of variables $V = \{x_1, x_2, \ldots\}$. We define the set of λ-terms, which we denote by $\Lambda$ inductively:

Variables $V \subseteq \Lambda$
Abstraction If $x \in V$ and $M \in \Lambda$, then $(\lambda x \centerdot M) \in \Lambda$
Application If $M, N \in \Lambda$, then $(MN) \in \Lambda$.

We will always identify terms that are the same up to a renaming of bound variables ($\alpha$ equivalence). Moreover, we define a notion of convertibility $=_{\beta\eta}$ formed by the reduction rules

$$(\lambda x \centerdot M)N \to_\beta M[x := N]$$

$$(\lambda x \centerdot Mx) \to_\eta M \quad \text{variable } x \text{ does not appear free in } M$$

Using this notion of reduction, a term $M$ is in $\beta\eta$ *normal form* if there is no term $N$ such that $M \to_{\beta\eta} N$.

We write $\Lambda^{\emptyset}$ for the set of all closed terms, that is, terms with no free variables.

*Remark 1* (Notational Conventions for Terms) We will follow all ordinary conventions when writing terms. When parentheses are omitted, we will associate terms to the left, so that $MNP$ is parsed as $(MN)P$. The symbol $\centerdot$ which is "Church's dot notation" will be used for binders to remind the reader that we are binding in the largest scope possible; so in $\lambda x \centerdot MN$ the $x$ is bound by the λ in both $M$ and $N$. We write λ-terms with uppercase Latin letters, like $F$, $G$, $M$, $N$.

What follows are a few useful closed terms that we will use throughout the paper:

$$S := \lambda xyz \centerdot xz(yz)$$

$$K = \textbf{True} := \lambda xy \centerdot x$$

$$K^* = \textbf{False} := \lambda xy \centerdot y$$

$$\textbf{And} := \lambda mn \centerdot \lambda xy \centerdot m(nxy)y$$

$$\textbf{Not} := \lambda m \centerdot \lambda xy \centerdot myx$$

## Simple Types

Here, we will put a typing discipline on $\Lambda$.

**Definition 2** On $\Lambda$ we define the simple typing system in the style of Church, which we notate $\lambda_{\rightarrow}$. We will have only one type constant, which we denote 0. The set of types, $T$, is inductively defined as the smallest set containing 0 and closed under arrows, that is, if $\alpha$ and $\beta$ are types, then $\alpha \rightarrow \beta$ is a type. We will use lower case Greek letters at the beginning of the alphabet for types, such as $\alpha, \beta, \gamma$.

For the terms of the system, we mimic what we did the untyped case, but with restrictions. We defined the set $\Lambda_{\rightarrow}(\alpha)$, the set of typed terms of type $\alpha$, by induction:

Variables If $x \in V$ and $\alpha$ is a type, then $x^{\alpha} \in \Lambda_{\rightarrow}(\alpha)$.
Abstraction If $M \in \Lambda_{\rightarrow}(\beta)$ and $x \in V$, then

$$\left(\lambda x^{\alpha} \centerdot M\right) \in \Lambda_{\rightarrow}(\alpha \rightarrow \beta)$$

Application If $M \in \Lambda_{\rightarrow}(\alpha \rightarrow \beta)$ and $N \in \Lambda_{\rightarrow}(\alpha)$ then

$$MN \in \Lambda_{\rightarrow}(\beta)$$

We write $\Lambda^{\emptyset}_{\rightarrow}(\alpha)$ for the set of closed terms of type $\alpha$. We write $\Lambda_{\rightarrow}$ for $\bigcup_{\alpha} \Lambda_{\rightarrow}(\alpha)$. Similarly for $\Lambda^{\emptyset}_{\rightarrow}$. Instead of writing $M \in \Lambda_{\rightarrow}(\alpha)$, we will usually write $M : \alpha$, which we read "$M$ has type $\alpha$" or "$M$ is in $\alpha$."

*Example 1* The closed terms we had previously defined are all typeable. For example, one can see both **True** and **False** have types $0 \rightarrow (0 \rightarrow 0)$.

**Definition 3** (Long Normal Form) We define the long ($\beta\eta$) normal form of a term $M : \alpha$ by induction on the type $\alpha$. If $M : 0$, then $M$ is in long normal form if and only if it is of the form $xM_1 \dots M_m$ where each $M_i$ is in long normal form. If $M : \alpha \rightarrow \beta$, then $M$ is in long normal form if $M$ is of the form $\lambda f^{\alpha} \centerdot N$, where $N$ is in long normal form. Every term has a unique long normal form, which one can obtain by $\eta$ expansions.

*Example 2* The long normal form of the term $\lambda x^0 \centerdot f^{0\rightarrow(0\rightarrow 0)}x$ is

$$\lambda x^0 \lambda y^0 \centerdot \left(f^{0\rightarrow(0\rightarrow 0)}x\right)y$$

*Remark 2* (Notational Conventions for Types) We will associate types the right to facilitate Currying; therefore $\alpha \rightarrow \beta \rightarrow \gamma$ is parsed as $\alpha \rightarrow (\beta \rightarrow \gamma)$. We will tend to not decorate variables with types when it is otherwise deducible from context what type the variable is.

We will use the notation $\alpha^n \rightarrow \beta$ to stand for $(\alpha \rightarrow (\alpha \rightarrow \cdots (\alpha \rightarrow \beta) \cdots))$. That is, a term of this type expects $n$-many inputs of type $\alpha$ and returns an output of type $\beta$. Also to improve readability, we will write the type for booleans $\alpha \rightarrow \alpha \rightarrow \alpha$ as $Bool_{\alpha}$.

## 2.2 Semantics

In this section, we will define a set-theoretic framework for which we can interpret typed lambda terms. We first give some more general definitions because we will later introduce different semantics.

**Definition 4** Suppose we have an indexed family of sets $\mathscr{M}(\alpha)$ for each type $\alpha$. Let $\cdot_{\alpha,\beta}$ be a map from $\mathscr{M}(\alpha \to \beta) \times \mathscr{M}(\alpha) \to \mathscr{M}(\beta)$. We say that this is a *typed applicative structure* if it is extensional; that is, for every $f, g \in \mathscr{M}(\alpha \to \beta)$, if we know for every $n \in \mathscr{M}(\alpha)$ $f \cdot_{\alpha,\beta} n = g \cdot_{\alpha,\beta} n$, then $f = g$.

**Definition 5** Fix a natural number $n$. We define a model $\mathscr{M}^n$ as follows, by induction on the type $\alpha$:

$$\mathscr{M}^n(0) := \{1, \dots, n\}$$

$$\mathscr{M}^n(\alpha \to \beta) := \big\{ f \mid f : \mathscr{M}^n(\alpha) \to \mathscr{M}^n(\beta) \big\}$$

That is, we interpret the type 0 to be the set $[n]$, the first $n$ natural numbers, and the type $\alpha \to \beta$ is the function space of objects of type $\alpha$ to objects of type $\beta$. Note that this is clearly a typed applicative structure where $\cdot$ is just function application. This particular typed applicative structure is called the *full type structure over* $[n]$.

We will write these set-theoretic functions as lowercase Latin letters like $f, g, h$.

Now that we have a framework which to interpret types, we can make an evaluation of the terms into this framework.

**Definition 6** Fix a natural number $n$ and a function $\varphi$ mapping typed variables $x^\alpha$ to members of $\mathscr{M}^n(\alpha)$. Then we define the evaluation of term with respect to $\varphi$ by induction on the term:

$$[\![x]\!]^n_\varphi = \varphi(x)$$

$$[\![\lambda x \bullet M]\!]^n_\varphi = \mathbb{\lambda} f \bullet [\![M]\!]^n_{\varphi[x:=f]}$$

$$[\![MN]\!]^n_\varphi = [\![M]\!]^n_\varphi\big([\![N]\!]^n_\varphi\big)$$

## 2.3 Type Theory

To begin a path from simply typed lambda calculus to a type theory, we need an equality symbol, which we shall call $\delta$. Following the example of Andrews [1], such an addition for all types would lead us to the study of higher order logic. For our purposes, we will just be dealing with first order (classical) logic, and our equality symbol is only for ground type 0.

To add equality, we augment our language with a new constant symbol $\delta$. For the typing rules, we just say that $\delta$ is a term of type $0 \to 0 \to \mathrm{Bool}_0$, defined by the following axiom:

$$(x = y \implies \delta xyuv = u) \wedge \big(\neg(x = y) \implies \delta xyuv = v\big)$$

In Statman [7] it was proven that under $\beta\eta$ conversion, the equational consequences of this axiom are exactly the same as from these rules:

$$\begin{aligned}
\delta MMUV &= U && \text{(Reflexivity)}\\
\delta MNUU &= U && \text{(Identity)}\\
\delta XYUV &= \delta YXVU && \text{(Symmetry)}\\
\delta XYXY &= Y && \text{(Hypothesis)}\\
P(\delta MN) &= \delta MN(P\mathbf{True})(P\mathbf{False}) && \text{(Monotonicity)}\\
\delta MN(\delta MNUV)W &= \delta MNUW && \text{(Stutter)}\\
\delta MNU(\delta MNWV) &= \delta MNUV && \text{(Stammer)}
\end{aligned}$$

In the previous section on the untyped lambda calculus, we define closed terms representing the booleans of **True** and **False**, as well as **And** and **Not**. For first-order type theory we must add a first-order quantifier, $\exists : (0 \to \text{Bool}_0) \to \text{Bool}_0$ with the rule

$$\exists M = \begin{cases} \mathbf{True} & \text{if } Mn = \mathbf{True} \text{ for some } n : 0 \\ \mathbf{False} & \text{otherwise} \end{cases}$$

and a description operator $\iota$ of type $(0 \to \text{Bool}_0) \to 0 \to 0$ with the rule

$$\iota Mm = \begin{cases} n & \text{if } Mn = \mathbf{True} \text{ and } n \text{ is unique such} \\ m & \text{otherwise} \end{cases}$$

Also, we can extend our semantics to handle the terms involving $\delta, \exists$, and $\iota$ in the obvious way; for example, for the equality operator, $[\![\delta]\!]^n_\varphi$ is the characteristic function of equality. The following shows that the semantics provided by $\mathcal{M}(n)$ is sound and complete for $\beta\eta\delta$.

**Theorem 1** (Soundness and Completeness) *Let $M$ and $N$ be terms of type $\alpha$.*

1. (*Soundness*) *If $M =_{\beta\eta\delta} N$, then for every $n \in \mathbb{N}$ and every $\varphi$, we have $[\![M]\!]^n_\varphi = [\![N]\!]^n_\varphi$*
2. (*Completeness*) *If $M \neq_{\beta\eta\delta} N$, then there are some $n \in \mathbb{N}$ and $\varphi$ such that $[\![M]\!]^n_\varphi \neq [\![N]\!]^n_\varphi$*

*Proof* Proof in Statman [7]. □

## 3 Henkin's Theorem

We would like to be able to say that every finite function in the above semantics can be represented in some way in the $\lambda\delta$-calculus.

**Theorem 2** (Henkin's Theorem) *Fix an assignment of variables to types $\varphi$, a natural number $n$, and a type $\alpha$.*

- *There is a* $\delta_\alpha : 0^n \to \alpha \to \alpha \to \text{Bool}_\alpha$ *such that for every* $f, g, h, j \in \mathscr{M}^n(\alpha)$,

$$\left([\![\delta_\alpha]\!]^n_\varphi 1 \cdots n\right) fghj = \begin{cases} h & \text{if } f = g \\ j & \text{if } f \neq g \end{cases}$$

- *If* $f \in \mathscr{M}^n(\alpha)$, *then there is* $F : 0^n \to \alpha$ *such that*

$$\left([\![F]\!]^n_\varphi 1 \cdots n\right) = f$$

*Proof* Go by induction on the type $\alpha$.

If $\alpha = 0$, then define

$$\delta_0 = \lambda x_1 \ldots x_n \bullet \delta$$

If $f \in \mathscr{M}^n(0)$, then $f \in [n]$, so $f = i$ for some $1 \leq i \leq n$. Then we can just $F$ be the $i$th projection:

$$\lambda x_1 \ldots x_n \bullet x_i$$

Suppose $\alpha = \beta \to \gamma$. We have $\delta_\beta$ and $\delta_\gamma$, closed terms of types $\beta$ and $\gamma$, respectively, that have the desired property. Enumerate all elements of $\mathscr{M}^n(\beta)$, $\{m_1, m_2, \ldots, m_k\}$. By induction hypothesis we know that these are representable; that is, there are closed terms $M_1, \ldots, M_k$ such that $[\![M_i]\!]^n_\varphi 1 \ldots n = m_i$ for every $i$. So define

$$\begin{aligned} \delta_{\beta\to\gamma} = \lambda \bar{x} F G \bullet \\ & \delta_\gamma \bar{x}\big(F(M_1\bar{x})\big)\big(G(M_1\bar{x})\big) \wedge \delta_\gamma \bar{x}\big(F(M_2\bar{x})\big)\big(G(M_2\bar{x})\big) \\ & \qquad \wedge \cdots \wedge \delta_\gamma \bar{x}\big(F(M_k\bar{x})\big)\big(G(M_k\bar{x})\big) \end{aligned}$$

where $\bar{x}$ is shorthand for $x_1 \ldots x_n$, and $M \wedge N$ is $\mathbf{And}(M)(N)$. One can easily verify this as desired.

Let $f$ be a function in $\mathscr{M}^n(\beta \to \gamma)$. Note that $f(m_i) \in \mathscr{M}^n(\gamma)$ by the definition of the semantics. Therefore, set $p_i = f(m_i)$. By induction hypothesis, there are closed terms $P_i$ for every $1 \leq i \leq k$ such that $([\![P_i]\!]^n_\varphi 1 \ldots n) = p_i$. We define $F$ by doing cases on what our input is:

$$\begin{aligned} F = \lambda \bar{x} m \bullet & \text{If } \delta_\beta \bar{x}(m)(M_1\bar{x}) \text{ then } (P_1\bar{x}) \text{ else} \\ & \quad \text{If } \delta_\beta \bar{x}(m)(M_2\bar{x}) \text{ then } (P_2\bar{x}) \text{ else} \\ & \qquad \ldots \\ & \qquad \text{If } \delta_\beta \bar{x}(m)(M_{k-1}\bar{x}) \text{ then } (P_{k-1}\bar{x}) \text{ else } (P_k\bar{x}) \end{aligned}$$

where "If $M$ then $N$ else $P$" is "shorthand" for $(MN)P$. Similarly, this is easily shown to be as claimed. □

The following is an easy corollary to the completeness result stated above and Henkin's theorem.

**Corollary 1** *Take $M, N : \alpha_1 \to \alpha_2 \to \cdots \to \alpha_k \to 0$ with all free variables having type* 0. *If $M \neq_{\beta\eta\delta} N$, then there are some $n$ and closed terms $F_i : 0^n \to \alpha_i$ such that*

$$M(F_1\bar{x})\cdots(F_k\bar{x}) \neq_{\beta\eta\delta} N(F_1\bar{x})\cdots(F_k\bar{x})$$

*where $\bar{x}$ includes all variables free in both $M$ and $N$.*

*Proof* By soundness, for some $\varphi$ and $n$, we have $[\![M]\!]^n_\varphi \neq [\![N]\!]^n_\varphi$. Take $\bar{x}$ to be a sequence of length $n$ of free variables of type 0, containing all the free variables of $M$ and $N$. Then clearly $[\![\lambda\bar{x}.M]\!]^n_\varphi \neq [\![\lambda\bar{x}.N]\!]^n_\varphi$.

These are set-theoretic functions, therefore there are $f_1 \in \mathscr{M}^n(\alpha_1), \ldots, f_k \in \mathscr{M}^n(\alpha_k)$ such that $([\![\lambda\bar{x}.M]\!]^n_\varphi 1\ldots n)f_1\ldots f_k \neq ([\![\lambda\bar{x}.N]\!]^n_\varphi 1\ldots n)f_1\ldots f_k$. By Henkin, all these $f_i$ have closed terms of type $0^n \to \alpha_i$ representing them; denote those closed terms $F_i$.

Therefore, by completeness, the result follows. □

Consider for each type $\alpha$ the set $\Lambda^\delta_\to(\alpha)$, that is, the set of terms of $\lambda^\delta_\to$ of type $\alpha$. Define the set $\mathscr{T}(\alpha)$ by

$$\mathscr{T}(\alpha) := \Lambda^\delta_\to(\alpha)/=_{\beta\eta\delta}$$

That is, all properly typed terms of type $\alpha$, modulo $\beta\eta\delta$ conversion. As a consequence to the above theorem, we have that $\mathscr{T}$ is an typed applicative structure. For each natural number $n$, we consider the set of $n$ free variables $X := \{x_1, x_2, \ldots, x_n\}$ of type 0. We can then take the set of all terms $M$ in $\mathscr{T}$ that are λ-definable from this set (and $\delta$).

This is not necessarily a typed applicative structure. For we may have two terms $M_1$ and $M_2$ that are not extensionally equal, but are with respect to all terms λ-definable from $X \cup \{\delta\}$. That is, none of the witnesses that $M_1$ and $M_2$ are different are λ-definable from $X \cup \{\delta\}$. Therefore, we consider only the equivalence classes formed by equality under $\delta$ restricted only to the ground set $X$. So, if we have $M_1$ and $M_2$ as above, we collapse them. We call the resulting model *Gandy hull* of $X \cup \{\delta\}$ in $\mathscr{T}$. This is a typed applicative structure, which we will denote by $\mathscr{T}^n$. For more information on the Gandy hull construction, see [2].

One can see that there is a natural relationship between $\mathscr{T}^n$ and $\mathscr{M}^n$. There is a natural homomorphism from $\mathscr{T}^n$ to $\mathscr{M}^n$, which is completely determined by a mapping of $X$ to $[n]$. Further, we can look at some infinite models. For instance, we can define $\mathscr{M}^\omega$ to be the full type structure over the natural numbers; so $\mathscr{M}^\omega(0) = \{1, 2, 3, \ldots\}$. We can then take the Gandy hull of $\{1, 2, 3, \ldots\} \cup \delta$ in this model and get a model $\mathscr{M}$.

This model could be obtained another way. Fix a bijection from free variables of type 0 and $\omega$. Then one can build a corresponding homomorphism from $\mathscr{T}$ to $\mathscr{M}^\omega$. The image is exactly $\mathscr{M}$. These models are discussed further in Statman [6].

## 4 Folklore Theorem

**Definition 7** The *symmetric group on n elements*, which we denote as $S_n$, is the subset of $\mathscr{M}^n(0 \to 0)$ that are bijections. These form a group with the operation of composition. We call members of the group permutations. We shall use lower case Greek letters in the middle of the alphabet to stand for permutations, for example, $\pi, \rho, \sigma, \tau$.

Of course, members of $S_n$ act canonically of type 0 by application. But, we can lift this action to higher types. Consider $\pi \in S_n$. We define $\pi_\alpha \in \mathscr{M}^n(\alpha \to \alpha)$ by induction on $\alpha$. If $\alpha = 0$, then we just take $\pi_0(n) = \pi(n)$. If $\alpha = \beta \to \gamma$, then we define

$$\pi_\alpha(f) = \pi_\gamma \circ f \circ \pi_\beta^{-1}$$

Therefore, we have an action of $S_n$ on our entire model $\mathscr{M}^n$, where $\pi$ acts on $f : \alpha$ by $\pi_\alpha(f)$. For this action, we will write $\pi \cdot f$.

If $f \in \mathscr{M}^n$, then we denote the stabilizer of $f$ under this action $\mathrm{St}(f)$; that is, $\mathrm{St}(f)$ is the set of all permutations that fix $f$.

$$\mathrm{St}(f) := \{\pi \in S_n \mid \pi \cdot f = f\}$$

We call an $f \in \mathscr{M}^n$ *symmetric* if $\mathrm{St}(f) = S_n$, that is, $f$ is fixed under the action of all permutations.

*Remark 3* We can say that $S_n$ acts on $\mathscr{T}^n$ as well. Any permutation of the free variables elicits an automorphism on the entire set $\mathscr{T}^n$. The converse, however, is false; there are automorphisms of $\mathscr{T}^n$ that do not come from permutations of the variables. Therefore, when we call $F \in \mathscr{T}^n$ symmetric, we mean preserved under all automorphisms, not just the "inner" automorphisms arising from permutations of variables.

**Theorem 3** (Folklore Theorem) *$f \in \mathscr{M}^n$ is symmetric if and only if it is λ-definable from $\delta, \iota, \exists$.*

*Proof* The right-to-left direction is straightforward. For $\delta$, $\iota$, and $\exists$ are all symmetric, as are combinators $S$ and $K$. $S$ and $K$ form a basis for all λ terms, and λ-definable objects are closed under application. Thus, we have that all λ-definable objects are indeed symmetric.

The left-to-right direction will constitute the rest of this section of the paper.

**Definition 8** For each function $f : 0^n \to \alpha$, we associate a functional $f^+ : (0 \to 0) \to \alpha$ such that

$$f^+\pi = f(\pi 1)(\pi 2)\dots(\pi n)$$

A function $f$ is said to be *regular* if for every $g \in \mathscr{M}^n(0 \to 0)$ where $g \notin S_n$, we have $f^+g = g(1)$.

For the present moment, we will restrict our attention only on functions $f : 0^n \to 0$.

*Remark 4* Note that the action $\pi \cdot f$ in this case is the following:

$$\pi \cdot f = \lambda\!\!\lambda \bar{x} \centerdot \pi\big(f\big(\pi^{-1}x_1\big)\dots\big(\pi^{-1}x_n\big)\big)$$

This and that $\mathrm{St}(f)$ is a subgroup implies that $\pi \in \mathrm{St}(f)$ if and only if

$$\lambda\!\!\lambda \bar{x} \centerdot \pi^{-1}\big(f(\pi x_1)\dots(\pi x_n)\big) = f$$

**Definition 9** Fix an $f : 0^n \to 0$ regular. We define a relation $\sim_f$ on $S_n$ by

$$\pi \sim_f \sigma \iff \pi^{-1}\big(f\big(\pi(1)\big)\ldots\big(\pi(n)\big)\big) = \sigma^{-1}\big(f\big(\sigma(1)\big)\ldots\big(\sigma(n)\big)\big)$$

We restrict this relation to be a right congruence by taking its *right congruence hull*, which we denote $\sim_f^*$ and define by

$$\pi \sim_f^* \sigma \iff \forall \rho \in S_n \centerdot \pi\rho^{-1} \sim_f \sigma\rho^{-1}$$

**Lemma 1** *For $f : 0^n \to 0$ regular and $\pi \in S_n$, the following are equivalent*:

1. $\pi \in St(f)$.
2. *For all $\rho \in S_n$, we have $\pi\rho \sim_f \rho$.*
3. $\pi \sim_f^* \mathrm{id}$.

*Proof* ((1) ⇒ (2)). Take $\pi \in \mathrm{St}(f)$. By remark we have

$$\lambda\bar{x} \centerdot \pi^{-1}\big(f(\pi x_1)\ldots(\pi x_n)\big) = f$$

Fix $\rho \in S_n$. Apply $\rho(1), \rho(2), \ldots, \rho(n)$ to the left:

$$\pi^{-1}\big(f\big(\pi\rho(1)\big)\ldots\big(\pi\rho(n)\big)\big) = f\big(\rho(1)\big)\ldots\big(\rho(n)\big)$$

which gives us

$$\rho^{-1}\big(\pi^{-1}\big(f\big(\pi\rho(1)\big)\ldots\big(\pi\rho(n)\big)\big)\big) = \rho^{-1}\big(f\big(\rho(1)\big)\ldots\big(\rho(n)\big)\big)$$

which implies that $\pi\rho \sim_f \rho$.

((2) ⇒ (3)). Take $\rho \in S_n$. We want to show that $\pi\rho^{-1} \sim_f \mathrm{id}\,\rho^{-1}$. The right-hand side is of course just $\rho^{-1}$; therefore, this follows immediately from (2).

((3) ⇒ (1)). We want to show that

$$\lambda\bar{x} \centerdot \pi^{-1}\big(f(\pi x_1)\ldots(\pi x_n)\big) = f$$

By extensionality, it suffices to show that the above holds after an arbitrary application. Moreover, let us fix an arbitrary $g : 0 \to 0$ (not necessarily in $S_n$). The application of $g(1)$ to the left of both sides, followed by $g(2)$, etc., up to $g(n)$, is an arbitrary application as $g$ is arbitrary; thus, it suffices to show

$$\pi^{-1}\big(f\big(\pi g(1)\big)\ldots\big(\pi g(n)\big)\big) = f\big(g(1)\big)\ldots\big(g(n)\big)$$

If $g \notin S_n$, then by regularly of $f$, both sides are exactly $g(1)$. Otherwise, call $\rho := g$ is a member of $S_n$. By (3) (using the right congruence property on $\rho^{-1}$), we have that $\pi\rho \sim_f \rho$. This means that

$$\rho^{-1}\pi^{-1}\big(f\pi\big(\rho(1)\big)\ldots\pi\big(\rho(n)\big)\big) = \rho^{-1}\big(f\big(\rho(1)\ldots\rho(n)\big)\big)$$

which is exactly what we wanted. □

From the above one sees that $\sim_f^*$ partitions $S_n$ into equivalence classes, where $\mathrm{St}(f)$ is the class that contains id. All other classes can be written as unions of right cosets of the stabilizer, by property (2).

Let $\mathscr{B}$ be the set of equivalence classes. For each $B \in \mathscr{B}$, let $\chi_B : 0^n \to \mathrm{Bool}_0$ denote its characteristic function. That is,

$$\chi_B^+(\pi) = \begin{cases} \textbf{True} & \text{if } \pi \in B \\ \textbf{False} & \text{otherwise} \end{cases}$$

By the definition of the equivalence relation, if $\pi$ and $\sigma$ are in a block $B$, then $\pi^{-1}(f^+\pi) = \sigma^{-1}(f^+\sigma) =: i$. So when $f$ is given inputs corresponding to a permutation in $B$, $f$ is just the $i$th projection function. Thus, to define $f$, we need only know which block the given input it in. Therefore, $f$ itself is $\lambda\delta$-definable from the set $\{\chi_B \mid B \in \mathscr{B}\}$ via the function

$$\begin{aligned} F = \lambda x_1 \ldots x_n \centerdot \text{If } \chi_{B_1} x_1 \ldots x_n \text{ then } (x_{i_1}) \text{ else} \\ \text{If } \chi_{B_2} x_1 \ldots x_n \text{ then } (x_{i_2}) \text{ else} \\ \ldots \\ \text{If } \chi_{B_j} x_1 \ldots x_n \text{ then } (x_{i_j}) \text{ else } (x_1) \end{aligned}$$

where $\{B_1, \ldots B_j\} = \mathscr{B}$, and $i_k$ is the coordinate that $f$ projects on block $B_k$.

**Lemma 2** *If $f$ is regular, symmetric of type $0^n \to 0$, then $f$ is $\lambda\delta$-definable.*

*Proof* Since $f$ is symmetric, by (1), there is only one block of the equivalence class formed by $\sim_f^*$. Since $f$ is $\lambda\delta$-definable from the set of blocks, we have that $f$ is $\lambda\delta$-definable outright. □

Now, consider arbitrary symmetric $f : \alpha_1 \to \alpha_2 \to \cdots \to \alpha_k \to 0$. It suffices, given the above, to show that $f$ is definable from regular, symmetric functions of type $0^n \to 0$. Consider the set of lists

$$\mathscr{L} = \Lambda_{\to}^{\delta}(\alpha_1) \times \cdots \times \Lambda_{\to}^{\delta}(\alpha_k) = \big\{\langle f_1, f_2, \ldots, f_k\rangle \mid f_i : \alpha_i\big\}$$

For each list $L = \langle f_1, \ldots, f_k\rangle$ in $\mathscr{L}$, we define a function $c_L : 0^n \to 0$, which we call the *Lth coordinate function* defined by

$$c_L = \lambda x_1 \ldots x_n \centerdot \begin{cases} f(F_1 x_1 \ldots x_n) \ldots (F_k x_1 \ldots x_n) & \text{if } x_1, \ldots, x_k \text{ distinct} \\ x_1 & \text{otherwise} \end{cases}$$

where the $F_i : 0^n \to \alpha_i$ are the terms from Henkin's theorem corresponding to $f_i$. Each coordinate function is regular (by the cases defining it) and also symmetric (as $f$ is). So, each $c_L$ is $\lambda\delta$-definable. Thus, we need only show that $f$ is definable from its coordinate functions; $f$ is not definable outright, but we need to use $\iota$ and $\exists$. We begin by the remark that the function **alldiff** : $0^n \to \mathrm{Bool}_0$, which returns **True** if all the first $n$ inputs are different, and **False** otherwise is $\lambda\delta$-definable.

$$\textbf{alldiff} := \lambda x_1 \cdots x_n \centerdot \delta(x_1)(x_2)(\textbf{False})\big(\cdots \delta(x_{n-1})(x_n)(\textbf{False})(\textbf{True}) \cdots\big)$$

Now, we can define $f$:

$$f = \lambda x_1 \ldots x_k \boldsymbol{.} \iota \Bigg( \lambda z \boldsymbol{.} \exists y_1 \ldots y_n \boldsymbol{.} (\mathbf{alldiff}\, y_1 \ldots y_n) \wedge \bigvee_{\substack{L \in \mathscr{L} \\ L = \langle F_1, \ldots, F_k \rangle}} \big(\delta(x_1)(F_1 y_1 \ldots y_n) \wedge \ldots \wedge \delta(x_k)(F_k y_1 \ldots y_n) \wedge (c_L y_1 \ldots y_n x_1 \ldots x_k)(z)\big) \Bigg)$$

Therefore, we have that $f$ is definable if each of the $c_L$ is definable; the $c_L$ are regular functions of type $0^n$, which are therefore definable if they are symmetric. It is easy to see that if $f$ is symmetric, then so are its coordinate functions. Therefore, if $f$ is symmetric, then we can substitute the $\lambda\delta$-definition of $c_L$ into each of the $c_L$ above and get a $\lambda\delta$-definition of $f$ (using $\exists$ and $\iota$). □

Moreover, we can prove the following strengthening:

**Theorem 4** *Fix a function $f \in \mathscr{M}^n$ and $A \subseteq \mathscr{M}^n$. Then if $(\bigcap_{g \in A} St(g)) \subseteq St(f)$, then $f$ is $\lambda\delta$-definable from functions in $A$ along with $\iota, \exists$.*

*Proof* It is easy to see that $\mathrm{St}(f) = \bigcap \mathrm{St}(c_L)$, where the $c_L$ are the coordinate functions of $f$; for since $f$ and its coordinate functions are definable from each other, any permutation that fixes $f$ must fix its coordinate functions, and any that fixes all its coordinate functions fixes the function.

Therefore, it suffices that we prove the theorem only for $f : 0^n \to 0$ and, similarly, assume all $g \in A$ be of type $0^n \to 0$. We suppose that $(\bigcap_{g \in A} \mathrm{St}(g)) \subseteq \mathrm{St}(f)$. By Lemma 1, since $\mathrm{St}(g)$ is exactly the block of the equivalence relation $\sim_g^*$ containing id, it follows that the set of left cosets of $\bigcap_{g \in A} \mathrm{St}(g)$ is a finer partition of $S_n$ than the set of left cosets of $\mathrm{St}(f)$, which are exactly the blocks of $\sim_f^*$.

Therefore, on any left coset of $\bigcap_{g \in A} \mathrm{St}(g)$ we have that $f$ behaves like a projection operator since the coset is entirely contained in a block of $\sim_f^*$, which in turn is contained in a block of $\sim_f$. Thus, for any permutations $\pi$, we can identify the left coset of $\bigcap_{g \in A} \mathrm{St}(g)$ that $\pi$ is in. $f$ acts uniformly on that block as a projection function, so we can make a definition similar to the above definition of $f$ by its blocks in $\sim_f$. □

## 5 Definability and Symmetry

Let us return our attention to the term model $\mathscr{T}$, where members are terms with possible free variables among $x_1, x_2, \ldots$. We first state the following result of Lauchli [5].

**Proposition 1** (Lauchli) *There is a closed term $F \in \Lambda_\delta^\emptyset(\alpha)$ if and only if there is an $F \in \mathscr{T}(\alpha)$ symmetric* (*recall*: *for terms in $\mathscr{T}(\alpha)$, symmetric means fixed under all automorphisms*).

*Proof* In Lauchli [5], it is stated and proved in terms of intuitionist logic: $\vdash_I \alpha$ if and only if there is an "invariant" function of type $\alpha$. □

**Theorem 5** *Any $F \in \mathscr{T}$ is λδ-definable if and only if it is symmetric.*

*Proof* It is easy to see that every element of $\mathscr{T}$ that is λδ-definable is symmetric since it is $\beta\eta\delta$ equal to a closed term, which is fixed under all automorphisms. We will just prove the converse.

Let $F \in \mathscr{T}$ by symmetric; consider $F$ to be of type $\alpha_1 \to \cdots \to \alpha_k \to 0$. Write $F$ as $Gx_1 \dots x_n$, where $G$ is closed and free variables of $F$ are among $x_1, \dots, x_n$. By the proposition above, we can get a closed term $H : \alpha_1 \to \cdots \to \alpha_k \to 0$, which has the following form in long normal form:

$$\lambda y_1 \dots y_k \bullet H'$$

where $H'$ has type 0 and free variables only among $y_1, \dots y_k$. Consider

$$G \underbrace{H' \dots H'}_{n \text{ times}}$$

This is a term of type $\alpha_1 \to \cdots \to \alpha_k \to 0$, which has free variables only among $y_1, \dots y_k$. Thus, the term

$$M := \lambda y_1 \dots y_k \bullet G \underbrace{H' \dots H'}_{n \text{ many}} y_1 \dots y_k : \alpha_1 \to \cdots \to \alpha_k \to 0$$

is closed.

Recall that $F$ is symmetric. Therefore,

$$F = Gx_1 \dots x_n =_{\beta\eta\delta} Gy_1 \dots y_n$$

for any variables $y_i : 0$. Thus, by a substitution, we have that $F =_{\beta\eta\delta} GY_1 \dots Y_n$ for any $Y_i : 0$. Therefore, $M =_{\beta\eta\delta} F$ and is closed, thus is a λδ-definition of $F$. □

**Corollary 2** *Let $h : \mathscr{T} \to \mathscr{M}$ be defined as $x_i \mapsto i$. This is called the* canonical homomorphism. *A function $f \in \mathscr{M}$ is λδ-definable if and only if there is $F \in h^{-1}(f)$ symmetric.*

*Proof* Once again the forward direction is straightforward. For the backward direction, we just apply the last theorem. By the last theorem, if $F \in h^{-1}(f)$ is symmetric, then it is λδ-definable by some closed term $G$. Since $h(G) = f$ and $G$ is closed, $G$ is also a good λδ-definition for $f$. □

**Definition 10** We call a homomorphism $h : \mathscr{T}^n \to \mathscr{M}^m$ *canonical* if $x_i \mapsto i$ for all $1 \leq i \leq m$.

We say that an $F \in \mathscr{T}^n$ is *supersymmetric* if for every homomorphism $\varphi : \mathscr{T}^n \to \mathscr{T}^n$, $\varphi(F)$ is symmetric.

**Theorem 6** *$f \in \mathscr{M}^m$ is λδ-definable if and only if there is some $n > m$ and $F \in \mathscr{T}^n$ supersymmetric such that for all canonical homomorphisms $h : \mathscr{T}^n \to \mathscr{M}^m$, we have $h(F) = f$.*

*Proof* The left to right direction is trivial since $f$ being $\lambda\delta$-definable gives us a closed term that will satisfy all the requirements.

For the other direction, fix $f \in \mathscr{M}^m$ of type $\alpha_1 \to \alpha_2 \to \cdots \to \alpha_k \to 0$. Suppose that $n > m$ and $F \in \mathscr{T}^n$ is supersymmetric where for all homomorphisms $h : \mathscr{T}^n \to \mathscr{M}^m$, have $h(F) = f$. Write $F = F'x_1 \dots x_j$ where $F'$ is a closed term.

The idea is as follows: we will do induction on the number of free variables on $F$, $j$. We will construct a new term $M$ that has $j-1$ free variables and still has the property that it is supersymmetric and is sent to $f$ under all canonical homomorphisms. At the end of our construction, we will have eliminated all free variables and will have constructed a closed term $M$ that is sent to $f$ under all canonical homomorphisms. But, since $M$ will be closed, $M$ is a $\lambda\delta$-definition for $f$.

To start the induction, if $j = 1$, then $F = F'x_1$. As $n > m \geq 1$, we know $n > 1$, so that $x_n \neq x_1$. $F$ is supersymmetric, and therefore it is symmetric, so under the automorphism sending $x_1$ to $x_n$, we know $F'x_1 = F'x_n$. As $n > m$, we have freedom with our canonical homomorphism to send $x_n$ anywhere; in particular, for any $1 \leq s \leq m$, we can define canonical homomorphism $h$ where $h(x_n) = s$. Therefore, $f = F's$ for all $s$. Therefore, we may replace $x_1$ in $F$ by anything of type 0, and it would still be sent to $f$ through any canonical homomorphism.

By Lauchli [5], there is a closed term $G$ of type $\alpha_1 \to \cdots \alpha_k \to 0$. We can write $F$ as $\lambda z_1 \dots z_k {\bullet} F'x_1 z_1 \dots z_k$ by doing $\eta$ expansions. Then, replacing $x_1$ to form the term $\lambda z_1 \dots z_k {\bullet} F'(Gz_1 \dots z_k) z_1 \dots z_k$, we have a closed $\lambda\delta$ term equal to $f$.

If $j > 1$, then we wish to eliminate the variable $x_j$. If $j > m$, then we already have freedom to send $x_j$ to any number in a canonical homomorphism $h$. Therefore, for every $1 \leq s \leq m$, by picking a canonical homomorphism that sends $x_j$ to $s$ we have

$$f = h\big(F'x_1 \dots x_j\big) = F'1 \dots n\big(h(n+1)\big) \dots \big(h(j-1)\big)s$$

Since $s$ is unrestricted, we can replace $x_j$ with anything of type 0, and the above is still preserved. In particular, doing an $\eta$ expansion of $F$ gives us $F = \lambda z_1 \dots z_k {\bullet} F'x_1 \dots x_j z_1 \dots z_k$, and then replacing $x_j$, we get:

$$f = h\underbrace{\big(\lambda z_1 \dots z_k {\bullet} F'x_1 \dots x_{j-1}\big(F'x_1 \dots x_1 z_1 \dots z_k\big) z_1 \dots z_k\big)}_{M}$$

$M$ has only $j-1$ free variables. It remains to show that $M$ is supersymmetric. This, however, is not hard to see. Under the map $x_i \mapsto x_1$ for all $1 \leq i \leq j$, we have that, since $F$ is supersymmetric, $F'x_1 \dots x_1$ is symmetric and therefore preserved under all automorphisms. Therefore, for any homomorphism $\varphi : \mathscr{T}^n \to \mathscr{T}^n$, we will have $\varphi(M)$ symmetric since $\varphi(F)$ was symmetric and $M$ is just $F$ with a free variable replaced by a symmetric term.

If $1 < j \leq m < n$, we have by the symmetry of $F$ by the automorphism switching $x_j$ and $x_n$ that

$$F = F'x_1 \dots x_{j-1} x_n$$

Now, we have the freedom to send $x_n$ anywhere under any canonical homomorphism, and thus we can repeat what we did above to eliminate $x_n$. □

## References

1. Andrews, P.B.: General models, descriptions, and choice in type theory. J. Symb. Log. **37**(2), 385–394 (1972)
2. Barengregt, H., Dekkers, W., Statman, R.: Lambda Calculus with Types. Cambridge University Press, Cambridge (2011)
3. Church, A.: A formulation of the simple theory of types. J. Symb. Log. **5**(2), 56–68 (1940)
4. Henkin, L.: A theory of propositional types. Fundam. Math. **52**(3), 323–344 (1963)
5. Läuchli, H.: An abstract notion of realizability for which intuitionistic predicate calculus is complete. Stud. Logic Found. Math. **60**, 227–234 (1970)
6. Statman, R.: Completeness, invariance and lambda-definability. J. Symb. Log. **47**(1), 17–26 (1982)
7. Statman, R.: Church's lambda delta calculus. In: Logic for Programming and Automated Reasoning, pp. 293–307. Springer, Berlin (2000)
8. van Benthem, J., Doets, K.: Higher-order logic. In: Gabbay, D.M., Guenthner, F. (eds.) Handbook of Philosophical Logic, 2nd edn. vol. 1, pp. 189–244. Springer, Berlin (2001)

W. Gunther (✉) · R. Statman
Department of Mathematical Sciences, Carnegie Mellon University, Pittsburgh, PA 15213, USA
e-mail: wgunther@andrew.cmu.edu

R. Statman
e-mail: rs31@andrew.cmu.edu