Equality Between Functionals*

Harvey Friedman†

State University of New York at Buffalo

The λ-calculus, in both its typed and untyped forms, has primarily been regarded as an attempt to formalize the concept of **rule** or **process**, and ultimately to provide a new foundation for mathematics. It seems fair to say that this aspect of the λ-calculus is currently in an embryonic state of development, awaiting further conceptual advances.

There is, however, another aspect of the **typed** λ-calculus, which is readily understood. This is its connection with the full classical finite type structure over ω, (i.e., with the functionals of finite type). In an obvious way, each closed term in the typed defines a functional of finite type over ω. Call a functional **simple** if it is given by some closed term in the typed λ-calculus.

Several definitions of **convertibility** between terms in the λ-calculus with or without types, have been considered. The motivation for introducing these definitions has primarily been to analyze the notion of the **identity** between rules, or processes. The relation $\vdash s = t$ defined in the text, is equivalent to one of these definitions of convertibility. It is easy to see that any two convertible terms define the same functional of finite type. We show here that any two non-convertible terms define different functionals of finite type.[1] This, coupled with the known decidability of convertibility, tells us that "equality between simple functionals is recursive."

Let us call two functionals **strongly unequal** if they differ everywhere. We show that, in contrast to the above, "strong inequality between simple functionals is as complicated as the set of true sentences of type theory over ω."

Augment the typed λ-calculus by (primitive) recursion operators, and call the result the R-λ-calculus. The functionals denoted by closed R-terms are the primitive recursive functionals of finite type. We conclude the paper by demonstrating that "equality between primitive recursive functionals is complete Π^1_1."

*This research was partially supported by NSF P038823.

†Logic Colloquium, 1975 - Springer. **Typesetter's note:** Edited and re-typeset in December 2017. Some notation is changed and proofs with equality are divided into multiple lines.

[1] This result was obtained in 1970.

1 Typed λ-calculus

We first describe the syntax of the typed λ-calculus. The type symbols are given by

i. 0 is a type symbol
ii. if σ, τ are type symbols, then $\sigma \to \tau$ is a type symbol.

The variables are written x_n^σ. The terms s, their types, their sets of free variables $FV(s)$, and their sets of bound variables $BV(s)$ are given by

i. x_n^σ is a term of type σ, such that $FV(x_n^\sigma) = \{x_n^\sigma\}$ and $BV(x_n^\sigma) = \varnothing$.
ii. if s is a term of type $\sigma \to \tau$, and t a term of type σ, then $s\,t$ is a term of the type τ, such that $FV(s\,t) = FV(s) \cup FV(t)$ and $BV(s\,t) = BV(s) \cup BV(t)$.
iii. if s is a term of type τ, y a variable of type σ, then $\lambda y.s$ is a term of type $\sigma \to \tau$, such that $FV(\lambda y.s) = FV(s) \setminus \{y\}$ and $BV(\lambda y.s) = BV(s) \cup \{y\}$.

The collection of all terms is denoted by Tm.

We now describe the semantics we will use for the typed λ-calculus.

A **pre-structure** is a system $(\{D^\sigma\}, \{A_{\sigma\tau}\})$, where

- D^σ is a non-empty set, for each type symbol σ
- $A_{\sigma\tau} : D^{\sigma\to\tau} \times D^\sigma \to D^\tau$, for each type symbols σ, τ
- We require the following extensionality condition:
  if $x, y \in D^{\sigma\to\tau}$ and $((\forall z \in D^\sigma)\ A_{\sigma\tau}(x, z) = A_{\sigma\tau}(y, z))$, then $x = y$.

An **assignment** is the system $(\{D^\sigma\}, \{A_{\sigma\tau}\})$ is a function f whose domain is the set of all variables, and such that $f(x_n^\sigma) \in D^\sigma$. The set of all assignments is denoted Asg. If y is a variable then f_α^y is given by $f_\alpha^y(x) = f(x)$ for $y \neq x$ and $f_\alpha^y(y) = \alpha$.

A **structure** is a system $(\{D^\sigma\}, \{A_{\sigma\tau}\}, \mathrm{Val})$ such that $(\{D^\sigma\}, \{A_{\sigma\tau}\})$ is a pre-structure, and $\mathrm{Val} : \mathrm{Tm} \times \mathrm{Asg} \to D^\sigma$, such that the following clauses hold:

i. $\mathrm{Val}(x_n^\sigma, f) = f(x_n^\sigma)$
ii. $\mathrm{Val}(s\,t, f) = A_{\sigma\tau}(\mathrm{Val}(s, f), \mathrm{Val}(t, f))$
iii. $(\forall \alpha \in D^\sigma)\ A_{\sigma\tau}(\mathrm{Val}(\lambda x.s, f), \alpha) = \mathrm{Val}(s, f_\alpha^x)$, where s is of type τ and x is of type σ.

Suppose $(\{D^\sigma\}, \{A_{\sigma\tau}\})$ is a pre-structure.
Then there is at most one function Val such that $(\{D^\sigma\}, \{A_{\sigma\tau}\}, \mathrm{Val})$ is a structure. Thus we sometimes refer to **structures** $(\{D^\sigma\}, \{A_{\sigma\tau}\})$, meaning that some Val, $(\{D^\sigma\}, \{A_{\sigma\tau}\}, \mathrm{Val})$ is a structure.

For any structure $(\{D^\sigma\}, \{A_{\sigma\tau}\}, \mathrm{Val})$, we write $(\{D^\sigma\}, \{A_{\sigma\tau}\}) \vDash s = t[f]$, just in case $\mathrm{Val}(s, f) = \mathrm{Val}(t, f)$. Write $(\{D^\sigma\}, \{A_{\sigma\tau}\}) \vDash s = t$ if and only if $(\{D^\sigma\}, \{A_{\sigma\tau}\}) \vDash s = t[f]$ for all assignments f. Below, we will often leave off the subscripts of $A_{\sigma\tau}$.

Let B be a non-empty set. We introduce the important example of a structure, T_B often referred to as the full type structure over B. $T_B = (\{B^\sigma\}, \{A_{\sigma\tau}\})$, where $B^0 = B$, $B^{\sigma\to\tau} = (B^\tau)^{B^\sigma}$, $A_{\sigma\tau} = x(y)$.

We wish to show that, for all infinite B, the relation $T_B \vDash s = t$ is decidable, and is the same for all infinite B. As an intermediate step, we establish a **completeness theorem** for the typed λ-calculus.

If s is a term, x a variable, t a term of the same type as x, then let s^x_t denote the substitution of the term t for each free occurrence of x in s. This may be inductively defined in

i. $x^x_t = t$
ii. $y^x_t = y$, for variables $y \neq x$
iii. $(r\ s)^x_t = r^x_t\ s^x_t$
iv. $(\lambda x.s)^x_t = \lambda x.s$
v. $(\lambda y.s)^x_t = \lambda y.s^x_t$, for variables $y \neq x$

A **substitution** is a function g from all variables into terms, such that $g(x)$ has the same type as x. Similarly, let $s(g)$ denote the simultaneous substitution of each free occurrence of each variable y in s, by $g(y)$.

We now introduce **axioms** and **rules** to the typed λ-calculus.

1. $\lambda x.s = \lambda y.s^x_y$, if $y \notin FV(s) \cup BV(s)$
2. $(\lambda x.s)t = s^x_t$, if $BV(s) \cap FV(t) = \varnothing$
3. $\lambda x.s\ x = s$, if $x \notin FV(s)$
4. $s = s$, $\dfrac{s = t}{t = s}$, $\dfrac{s = t \qquad t = r}{s = r}$
5. $\dfrac{s = t}{r\ s = r\ t}$, $\dfrac{s = t}{s\ r = t\ r}$, $\dfrac{s = t}{\lambda x.s = \lambda x.t}$

We first prove the soundness theorem. Fix a structure $M = (\{D^\sigma\}, \{A_{\sigma\tau}\})$.

Lemma 1. $\text{Val}(s^x_t, f) = \text{Val}(s, f^x_{\text{Val}(t,f)})$, if $BV(s) \cap FV(t) = \varnothing$.

Proof: Fix t, and use induction on s. □

Lemma 2. $\text{Val}(s^x_y, f^y_\alpha) = \text{Val}(s, f^x_\alpha)$, if $y \notin FV(s) \cup BV(s)$.

Proof: From Lemma 1. □

Lemma 3. $M \vDash \lambda x.s = \lambda y.s^x_y$, if $y \notin FV(s) \cup BV(s)$.

Proof:
$$\begin{aligned} A(\text{Val}(\lambda x.s, f), \alpha) &= \text{Val}(s, f^x_\alpha) \\ &= \text{Val}(s^x_y, f^x_\alpha) \\ &= A(\text{Val}(\lambda y.s^x_y, f), \alpha), \text{ by Lemma 2. } \square \end{aligned}$$

Lemma 4. $\mathrm{Val}((\lambda x.s)t, f) = \mathrm{Val}(s^x_t, f)$, if $BV(S) \cap FV(t) = \varnothing$.

PROOF:
$$\begin{aligned}\mathrm{Val}((\lambda x.s)t, f) &= A(\mathrm{Val}(\lambda x.s, f), \mathrm{Val}(t, f)) \\ &= \mathrm{Val}(s, f^x_{\mathrm{Val}(t,f)}) \\ &= \mathrm{Val}(s^x_t, f), \text{ by Lemma 1. } \square\end{aligned}$$

Lemma 5. $\mathrm{Val}(\lambda x.s\ x, f) = \mathrm{Val}(s, f)$, if $x \notin FV(s)$.

PROOF:
$$\begin{aligned}A(\mathrm{Val}(\lambda x.s\ x, f), \alpha) &= \mathrm{Val}(s\ x, f^x_\alpha) \\ &= A(\mathrm{Val}(s, f^x_\alpha), \mathrm{Val}(x, f^x_\alpha)) \\ &= A(\mathrm{Val}(s, f), \alpha)\ \square\end{aligned}$$

Lemma 6. If $M \vDash s = t$ then $M \vDash \lambda x.s = \lambda x.t$.

PROOF: Fix f. Then
$$\begin{aligned}A(\mathrm{Val}(\lambda x.s, f), \alpha) &= \mathrm{Val}(s, f^x_\alpha) \\ &= \mathrm{Val}(t, f^x_\alpha) \\ &= A(\mathrm{Val}(\lambda x.t, f), \alpha)\ \square\end{aligned}$$

Theorem 1. (Soundness). If $\vdash s = t$, then every structure $M \vDash s = t$.

PROOF: By induction on the proof of $s = t$, using Lemmas 1, 2, 3, 4, 5 and 6. $\square$

For the proof of completeness, we consider a particular structure defined from the relation $\vdash$. Let $[s]$, for terms s, be $\{t \mid\ \vdash s = t\}$. It is clear that the $[s]$ are the equivalence classes of the equivalence relation $\vdash s = t$. This is because of the rules 4.

We wish to define a specific $M_0 = (\{D^\sigma\}, \{A_{\sigma\tau}\})$. Take $D^\sigma = \{[s] \mid s \text{ is of type } \sigma\}$. Define $A_{\sigma\tau} = [s\ t]$, where s is of type $\sigma \to \tau$, t is of type τ.

We must now check that M_0 is well-defined. Firstly we remark that if $\vdash s = t$, then s and t are of the same type. Secondly, note that if $\vdash s = s'$, $\vdash t = t'$, then $\vdash s\ t = s'\ t'$, by rules 5.

Lemma 7. M_0 is a pre-structure.

PROOF: Let $[s], [t] \in D^{\sigma\to\tau}$. Suppose that for all $r \in D^\sigma$, we have $[s\ r] = [t\ r]$. Then let x be a variable not free in either s or t. We have $[s\ x] = [t\ x]$. Hence $\vdash s\ x = t\ x$. By rule 5, $\vdash \lambda x.s\ x = \lambda x.t\ x$. By axiom 3, $\vdash \lambda x.s\ x = s$, $\vdash \lambda x.t\ x = t$. By rule 4, $\vdash s = t$. Hence $[s] = [t]$. $\square$

Let us call a substitution g, **regular**, just in case for all variables x, y, $FV(g(x)) \cap BV(g(y)) = \varnothing$.

Lemma 8. Let b be a finite set of variables, s a term. Then there is a term t such that $\vdash s = t$, $FV(s) = FV(t)$, and $BV(t) \cap b = \varnothing$.

Proof: By successive applications of axiom 1 and rules 4. $\square$

Let f be an assignment for M_0, s a term. We wish to define $\mathrm{Val}(s, f)$. By Lemma 8, let g be a regular substitution such that each $f(x) = [g(x)]$. Choose t to be a term such that $\vdash s = t$, and $BV(t) \cap FV(f(x)) = \varnothing$, for all $x \in FV(t)$, again by Lemma 8. Set $\mathrm{Val}(s, f) = [t(g)]$. We must now show that $\mathrm{Val}(s, f)$ is well-defined.

Lemma 9. Suppose g_1, g_2 are regular substitutions such that $\vdash g_1(x) = g_2(x)$, for all variables x. Suppose $\vdash s = t$, $BV(s) \cap FV(g_1(x)) = BV(t) \cap FV(g_2(y)) = \varnothing$, for all $x \in FV(s), y \in FV(t)$. Then $\vdash s(g_1) = t(g_2)$.

Proof: By induction on the cardinality k of $FV(s) \cup FV(t)$. The case $k = 0$ is trivial. Let $k = n + 1$, and assume true for n. Let $x \in FV(s) \cup FV(t)$. Choose w of the same type as x, so that $w \notin FV(s) \cup FV(t) \cup BV(s) \cup BV(t)$, $w \notin FV(g_1(t)) \cup FV(g_2(z))$, for all $y \in FV(s)$, $z \in FV(t)$. We have $\vdash \lambda x.s = \lambda x.t$, and $\vdash \lambda x.s = \lambda w.s^x_w$, $\lambda x.t = \lambda w.t^x_w$. So $\vdash \lambda w.s^x_w = \lambda w.t^x_w$. By induction hypothesis, $\vdash (\lambda w.s^x_w)(g_1) = (\lambda w.t^x_w)(g_2)$. Let $h_1 = (g_1)^x_x$, $h_2 = (g_2)^x_x$. Then $\vdash \lambda w.(s(h_1))^x_w = \lambda w.(t(h_2))^x_w$. Hence $\vdash (\lambda w.(s(h_1))^x_w)g_1(x) = (\lambda w.(t(h_2))^x_w)g_2(x)$. So $\vdash (s(h_1))^x_{g_1(x)} = (t(h_t))^x_{g_t(x)}$. Hence $s(g_1) = t(g_2)$. $\square$

Lemma 10. $\mathrm{Val}(s, f)$ is well-defined, and $\mathrm{Val}(s, f) = \mathrm{Val}(t, f)$ if $\vdash s = t$.

Proof: Obvious from Lemma 9. $\square$

We now wish to show that M_0 is a structure. Write $f = [g]$, for M_0-assignments f, if g is a regular substitution and $f(x) = [g(x)]$.

Lemma 11. $\mathrm{Val}(x, f) = f(x)$, for variables x.

Proof: Let $f = [g]$. Then $\mathrm{Val}(x, f) = [x(g)] = [g(x)] = f(x)$. $\square$

Lemma 12. $\mathrm{Val}(s\, t, f) = A(\mathrm{Val}(s, f), \mathrm{Val}(t, f))$.

Proof: Let $f = [g]$. Choose s', t' so that $\vdash s = s'$, $\vdash t = t'$, $BV(s') \cap FV(g(x)) = BV(t') \cap FV(g(x)) = \varnothing$, for all $x \in FV(s') \cup FV(t')$. Then

$$\begin{aligned}\mathrm{Val}(s\, t, f) &= [(s'\, t')(g)]\\ &= [(s'(g)\, t'(g)]\\ &= A([s'(g)], [t'(g)])\\ &= A(\mathrm{Val}(s, f), \mathrm{Val}(t, f)) \ \square\end{aligned}$$

Lemma 13. $A(\mathrm{Val}(\lambda x.s, f), [t]) = \mathrm{Val}(s, f^x_{[t]})$.

Proof: Choose $f^x_{[t]} = [g]$. Let $\vdash s = s'$, $BV(s') \cap FV(g(y)) = \varnothing$, for all $y \in FV(s')$. Then $\mathrm{Val}(\lambda x.s, f) = [(\lambda x.s')(g)]$. Let $h = g^x_x$. Then $(\lambda x.s)(g) = \lambda x.s'(h)$. Note that $\vdash (\lambda x.s'(h))(t) = (\lambda x.s'(h))(g(x)) = s'(h)^x_{g(x)} = s'(g)$. Hence $A(\mathrm{Val}(\lambda x.s, f), [t]) = [s'(g)] = \mathrm{Val}(s, f^x_{[t]})$. □

Lemma 14. $M_0 = (\{D^\sigma\}, \{A_{\sigma\tau}\}, \mathrm{Val})$ is a structure.

Proof: By Lemmas 11, 12 and 13. □

Theorem 2. (Completeness). Let s, t be terms. The following are equivalent:

i. $\vdash s = t$
ii. for all structures M, $M \vDash s = t$
iii. $M_0 \vDash s = t$

Proof: By Theorem 1, we must simply show that $M_0 \vDash s = t$, implies $\vdash s = t$. Suppose $M_0 \vDash s = t$. Let $f = [g]$, where g is the identity map, and choose s', t' such that $\vdash s = s'$, $\vdash t = t'$, and $BV(s') \cap FV(s') = BV(t') \cap FV(t') = \varnothing$. Then $\mathrm{Val}(s, f) = [s']$, $\mathrm{Val}(t, f) = [t']$. Hence $[s'] = [t']$, and so $\vdash s' = t'$. Hence $\vdash s = t$, and we are done. □

Let $M = (\{D^\sigma\}, \{A_{\sigma\tau}\}, \mathrm{Val}_1)$, $N = (\{E^\sigma\}, \{B_{\sigma\tau}\}, \mathrm{Val}_2)$ be structures. A system $\{f_\sigma\}$ is called a **partial homomorphism** from M onto N just in case

i. each f_σ is a partial surjective map from D_σ onto E_σ
ii. $f_{\sigma\tau}(x)$ is the unique element of $E^{\sigma\to\tau}$ (if it exists) such that $f_\tau(A(x, y)) = B(f_{\sigma\tau}(x), f_\sigma(y))$, for all $y \in \mathrm{dom}(f_\sigma)$.

Note that $\{f_\sigma\}$ is determined by f_0, and this definition does not involve Val. The following lemma does.

Lemma 15. If $\{f_\sigma\}$ is a partial homomorphism from M onto N, and g is an M-assignment, h is an N-assignment, $f_\sigma(g(x)) = h(x)$ for variables x of type σ, then $f_\sigma(\mathrm{Val}_1(s, g)) = \mathrm{Val}_2(s, h)$, for terms s of type σ.

Proof: By induction on s, where $M, N, \{f_\sigma\}$ are fixed.

$f_\sigma(\mathrm{Val}_1(x, g)) = f_\sigma(g(x)) = h(x) = \mathrm{Val}_2(x, h)$, for variables x of type σ.

Now $f_\tau(\mathrm{Val}_1(s\ t, g)) = f_\tau(A(\mathrm{Val}_1(s, g), \mathrm{Val}_1(t, g)))$. By induction hypothesis, $f_{\sigma\tau}(\mathrm{Val}_1(s, g)) = \mathrm{Val}_2(s, h)$ and $f_\sigma(\mathrm{Val}_1(t, g)) = \mathrm{Val}_2(t, h)$. Hence

$$\begin{aligned} f_\tau(A(\mathrm{Val}_1(s, g)), \mathrm{Val}_1(t, g)) &= B(\mathrm{Val}_2(s, h), \mathrm{Val}_2(t, h)) \\ &= \mathrm{Val}_2(s\ t, h) \end{aligned}$$

Finally we must show $f_{\sigma\tau}(\text{Val}_1(\lambda x.s, g)) = \text{Val}_2(\lambda x.s, h)$. To do this, let $y \in \text{dom}(f_\sigma)$. We must show $f_\tau(A(\text{Val}_1(\lambda x.s, g), y)) = B(\text{Val}_2(\lambda x.s, h), f_\sigma(y))$.

$$\begin{aligned} f_\tau(A(\text{Val}_1(\lambda x.s, g), y)) &= f_\tau(\text{Val}_1(s, g^x_y)) \\ &= \text{Val}_2(s, h^x_{f_\sigma(y)}) \\ &= B(\text{Val}_2(\lambda x.s, h), f_\sigma(y)) \ \square \end{aligned}$$

Lemma 16. Suppose there is a partial homomorphism from M onto N. Then $M \vDash s = t$ implies $N \vdash s = t$, for any terms s, t.

Proof: Let $\{f_\sigma\}$ be a partial homomorphism, and assume $M \vDash s = t$. Let h be an N-assignment. Choose an M-assignment g so that $h(x) = f_\sigma(g(x))$, for variables x of type σ. Then $\text{Val}_2(s, h) = f_\sigma(\text{Val}_1(s, g)) = f_\sigma(\text{Val}_1(t, g)) = \text{Val}_2(t, h)$, by Lemma 15, for terms s, t of type σ. Hence $N \vDash s = t$. $\square$

For sets B, let $|B|$ be the cardinal of B.

Lemma 17. Let $M = (\{D^\sigma\}, \{A_{\sigma\tau}\}, \text{Val})$ be a structure $|D^0| \leq |B|$. Then there is a partial homomorphism from T_B onto M.

Proof: We define $\{f_\sigma\}$ by induction on the type symbol σ. Let f_0 be any partial surjective map from B onto D^0. Suppose f_σ, f_τ have been defined, surjectively, according to the clauses for being a partial homomorphism. Define $f_{\sigma\tau}(x)$ to be the unique element of $D^{\sigma\to\tau}$ (if it exists) such that $f\tau(x(y)) = A(f_{\sigma\tau}(x), f_\sigma(y))$, for all $y \in \text{dom}(f_\sigma)$. We must show that $f_{\sigma\tau}$ is surjective. Let $z \in D_{\sigma\to\tau}$. Choose $x \in B^{\sigma\to\tau}$ so that for all $y \in \text{dom}(f_\sigma)$, $x(y) \in f_\tau^{-1}(A(z, f_\sigma(y)))$. Then $f_{\sigma\tau}(x) = z$. $\square$

Theorem 3. (Extended Completeness). Let s, t be terms. B an infinite set. The following are equivalent:

i. $\vdash s = t$
ii. for all structures M, $M \vDash s = t$
iii. $T_B \vDash s = t$

Proof: By Theorem 2, it suffices to show that $T_B \vDash s = t$ implies $M_0 \vDash s = t$. By Lemma 17, there is a partial homomorphism from T_B onto M. By Lemma 16, if $T_B \vDash s = t$, then $M \vDash s = t$. $\square$

Lemma 18. The relation $\vdash s = t$ is recursive.

Proof: This follows from the following known fact about the typed λ-calculus (even with recursion operators): every term reduces to a unique irreducible term, up to changes in bound variables, no matter how the reductions are performed (see Sanchis[1], Tait[2], and Barendregt[3] for elaboration). $\square$

Corollary. If B is infinite then the relation $T_B \vDash s = t$ is recursive, and is independent of the size of B.

Let B be finite, $g : B \to B$. Define $g^1 = g$, $g^{k+1} = g \circ g^k$. We will show that the extended completeness theorem fails for B.

Lemma 19. There are $i(g), j(g) \geq 1$ such that for $n > 1$, we have $g^n = g^{n+j}$.

Fix x to be a variable of type $0 \to 0$. Let $x^1 = x$, $x^{k+1} = x \circ x^k = \lambda y.x(x^k(y))$.

Theorem 4. For each non-empty finite B, there are terms s, t of type $0 \to 0$ with $T_B \vDash s = t$, such that not $\vdash s = t$.

Proof: For each $g : B \to B$ define $i(g), j(g)$ as in Lemma 19. Choose i greater than each $i(g)$, and set $j = \Pi j(g)$. Then for each $g : B \to B$ we have $g^i = g^{i+j}$. Hence $T_B \vDash x^i = x^{i+j}$. To see that not $\vdash s = t$, consider T_ω. Note that not $T_\omega \vDash x^i = x^{i+j}$, since x may be interpreted as the successor function on ω. We are done. $\square$

Now let $M \vDash s \neq t$ mean not $M \vDash s = t[f]$, for all M-assignments f. (We will often write $M = s \neq t[f]$ for not $M \vDash s = t[f]$. Does corollary to Theorem 3 hold for relation $T_B \vDash s \neq t$? Below, we give a negative answer.

We introduce a many-sorted predicate calculus (with equality), $\mathcal{L}$, appropriate for the theory of functionals of finite type over a nonempty domain. Specifically, the **atomic formulae** of $\mathcal{L}$ are written $s = t$, for terms s, t, of the types λ-calculus of the same type. The **formulae** of $\mathcal{L}$ are obtained from the atomic formulae by using $\neg, \wedge, \forall$. The $\forall$-quantifiers quantify over a given type only. Thus $M \vDash \varphi[f]$, for structures M, formulae φ of $\mathcal{L}$, and M-assignments f, is defined in the obvious way. $\exists, \vee$ are introduced as abbreviations in the standard manner. Take $M \vDash \varphi$ to mean $M \vDash \varphi[f]$ for all M-assignments f.

A formula φ of $\mathcal{L}$ is called **existential** if it is of the form $(\exists x)(s = t)$.

Let 0 be the closed term $\lambda y.\lambda x.y$, 1 the closed term $\lambda y.\lambda x.x$, where x, y are distinct variables of type 0. For terms s, t, let $\langle s, t\rangle$ be the term $\lambda x.(x\ s)\ t$, where $x \notin FV(s) \cup FV(t)$, so that $\langle s, t\rangle$ has type 0.

Lemma 20. If B has at least two elements, then $T_B \vDash 0 \neq 1$.

Lemma 21. If s, u have the same type, t, v have the same type, then $T_B \vDash \langle s, t\rangle = \langle u, v\rangle \to (s = u \wedge t = v)$.

Lemma 22. If φ is existential, there is existential ψ with the same free variables, such that $T_B \vDash \psi \leftrightarrow \neg\varphi$, for all B with at least two elements.

Proof: Let φ be $(\exists x)(s = t)$. Note that by Lemma 20, $T_B \vDash (\exists y)(y\ s = 0 \wedge y\ t = 1) \leftrightarrow s \neq t$, where $y \notin FV(s) \cup FV(t)$, $y \neq x$. By Lemma 21, $T_B \vDash (\exists y)(\langle y\ s, y\ t\rangle = \langle 0, 1\rangle) \leftrightarrow s \neq t$. Hence $T_B \vDash (\forall x)(\exists y)(\langle y\ s, y\ t\rangle = \langle 0, 1\rangle \leftrightarrow (\forall x)(s \neq t)$. So $T_B \vDash (\exists z)(\forall x)(\langle (z\ x)\ s, (z\ x)\ t\rangle = \langle 0, 1\rangle) \leftrightarrow (\forall x)(s \neq t)$, where $x \notin FV(s) \cup FV(t)$, $x \neq x, y$. Hence $T_B \vDash (\exists z)((\lambda x.\langle (z\ x)\ s, (z\ x)\ t\rangle) = \lambda x.\langle 0, 1\rangle) \leftrightarrow (\forall x)(s \neq t)$. $\square$

Lemma 23. If φ, ψ are existential, then there is an existential ρ with the same free variables as $\varphi \wedge \psi$, such that $T_B \vDash \rho \leftrightarrow (\varphi \wedge \psi)$, for all B with at least two elements.

Proof: Let φ be $(\exists x)(s = t)$, ψ be $(\exists x)(u = v)$, (where φ, ψ may have had their bound variable changed to x). Then $T_B \vDash (\exists x)(\langle s, u\rangle = \langle t, v\rangle) \leftrightarrow ((\exists x)(s = t) \wedge (\exists x)(u = v))$. $\square$

Lemma 24. If φ is existential, then there is an existential ρ with the same free variables as $(\exists x)(\varphi)$, such that $T_B \vDash \rho \leftrightarrow (\exists x)(\varphi)$, for all B with at least two elements.

Proof: Let φ be $(\exists y)(s = t)$, where $y \neq x$, (where φ may have had its bound variable changed). Then $T_B \vDash (\exists z)((s^x_{z\ 0})^y_{z\ 1} = (t^x_{z\ 0})^y_{z\ 1}) \leftrightarrow (\exists x)(\exists y)(s = t)$. $\square$

Lemma 25. For each formula φ of $\mathcal{L}$, we can effectively find an existential ψ with the same free variables, such that $T_B \vDash \varphi \leftrightarrow \psi$, for each B with at least two elements.

Proof: From Lemmas 22, 23 and 24. $\square$

Lemma 26. There is a one-one total recursive function f such that for each formula φ of $\mathcal{L}$, $f(\varphi)$ is an existential formula with the same free variables as φ, and $T_B \vDash \varphi \leftrightarrow \psi$, for each B with at least two elements.

Proof: This is an effective version of Lemma 25, obtained from corresponding effective version of Lemmas 22, 23 and 24. $\square$

Theorem 5. For each B, the set of sentences φ of $\mathcal{L}$ such that $T_B \vDash \varphi$, is one-one reducible to the relation $T_B \vDash s \neq t$.

Proof: We can assume that B has at least two elements (or for that matter, is infinite), since otherwise $\{\varphi : T_B \vDash \varphi\}$ is recursive. Note that for sentences φ of $\mathcal{L}$, $T_B \vDash \varphi$ if and only if not $T_B \vDash \neq \varphi$ if and only if not $T_B \vDash f(\neq \varphi)$ if and only if $T_B \vDash s \neq t$, where $f(\neg\varphi) = (\exists x)(s = t)$. $\square$

2 The typed λ-calculus with primitive recursion

We will refer to this extension of the typed λ-calculus as the R-λ-calculus. The R-λ-calculus has the additional symbols O,N, and R_σ for each type symbol σ. The variables of the R-λ-calculus are the same as the variables of the variables of the typed λ-calculus.

The **terms** s , their **types**, their sets of **free variables** $FV(s)$, and their sets of **bound variables** $BV(s)$ are given by

i. x_n^σ is a term of type σ, such that $FV(x_n^\sigma) = \{x_n^\sigma\}$ and $BV(x_n^\sigma) = \varnothing$.
ii. if s is a term of type $\sigma \to \tau$, and t a term of type σ, then $s\ t$ is a term of the type τ, such that $FV(s\ t) = FV(s) \cup FV(t)$ and $BV(s\ t) = BV(s) \cup BV(t)$.
iii. if s is a term of type τ, y a variable of type σ, then $\lambda y.s$ is a term of type $\sigma \to \tau$, such that $FV(\lambda y.s) = FV(s) \setminus \{y\}$ and $BV(\lambda y.s) = BV(s) \cup \{y\}$.
iv. 0 is a term of type 0, $FV(0) = \varnothing$, $BV(0) = \varnothing$.
v. N is a term of type $0 \to 0$, $FV(N) = \varnothing$, $BV(N) = \varnothing$.
vi. R_σ is a term of type $(\sigma \to 0 \to \sigma) \to \sigma \to 0 \to \sigma$, $FV(R_\sigma) = \varnothing$, $BV(R_\sigma) = \varnothing$.

We let $(s_1, ..., s_{n_1})$ be $((s_1\ s_2), ..., s_{n_1})$.

Let $(\{D^\sigma\}, \{A_{\sigma\tau}\})$ be a pre-structure. It wil be convenient to assume that D^σ are disjoint. Let $A(x_1, ..., x_{n_1})$, for appropriate $x_1, ..., x_{n_1}$ be $A(A(x_1, x_2), ..., x_{n+1})$, where each occurrence of A denotes the appropriate $A_{\sigma\tau}$.

A system $(\{D^\sigma\}, \{A_{\sigma\tau}\}, \mathrm{Val})$ is an **R-structure** just in case

i. $(\{D^\sigma\}, \{A_{\sigma\tau}\})$ is a pre-structure
ii. $D^0 = \omega$
iii. $\mathrm{Val}(x_n^\sigma, f) = f(x_n^\sigma)$
iv. $\mathrm{Val}(s\ t, f) = A(\mathrm{Val}(s, f), \mathrm{Val}(t, f))$
v. for all $\alpha \in D^\sigma$, $A(\mathrm{Val}(\lambda x.s, f), \alpha) = \mathrm{Val}(s, f_\alpha^x)$, where s is of type σ
vi. $\mathrm{Val}(0, f) = 0$
vii. $A(\mathrm{Val}(N, f), k) = k + 1$
viii. $A(\mathrm{Val}(R_\sigma, f), y, z, 0) = z$, $A(\mathrm{Val}(R_\sigma, f), y, z, k+1) = A(y, A(\mathrm{Val}(R_\sigma, f), y, z, k), k)$ for $y \in D^{\sigma\to(0\to\sigma)}$, $z \in D^\sigma$, $k \in \omega$.

Note that if $(\{D^\sigma\}, \{A_{\sigma\tau}\})$ is a pre-structure, there is at most one Val such that $(\{D^\sigma\}, \{A_{\sigma\tau}\}, \mathrm{Val})$ is an R-structure. Thus we may refer to the R-structure $(\{D^\sigma\}, \{A_{\sigma\tau}\})$.

Obviously, we may view T_ω as an R-structure just as we viewed T_ω as a structure in section 1.

As in section 1, we write $M \vDash s = t[f]$ to mean $\mathrm{Val}(s, f) = \mathrm{Val}(t, f)$, and $M \vDash s = t$ to mean $\mathrm{Val}(s, f) = \mathrm{Val}(t, f)$, for all assignments f. In this section we will show that the relation $T_\omega \vDash s = t$ is complete Π_1^1.

Let $M = (\{D^\sigma\}, \{A_{\sigma\tau}\}, \mathrm{Val}_1)$, $N = (\{E^\sigma\}, \{B_{\sigma\tau}\}, \mathrm{Val}_2)$ be R-structures. A system $\{f_\sigma\}$ is called a **partial homomorphism** from M onto N just in case it is one when M, N are viewed as structures (the definition did not involve Val), and f_0 is the identity.

Lemma 27. Suppose $\{f_\sigma\}$ is a partial homomorphism from M onto N. Suppose $x_1, ..., x_n$ are respectively $\mathrm{dom}(f\tau_1), ..., \mathrm{dom}(f_{\tau_n})$, and $A(x_1, ..., x_n) \in D^\sigma$. Then $f_\sigma(A(x_1, ..., x_n)) = B(f_{\tau_1}(x_1), ..., f_{\tau_n}(x_n))$.

Proof: By induction on n. For $n = 2$, this is straight from the definition of partial homomorphism. Using this, we have

$$\begin{aligned} f_\sigma(A(x_1, ..., x_{n+1}) &= f_\sigma(A(A(x_1, x_2), ..., x_{n+1})) \\ &= B(f_\tau(A(x_1, x_2)), f\tau_3(x_3), ..., f_{\tau_n}(x_n)) \\ &= B(B(f_{\tau_1}(x_1), f_{\tau_2}(x_2), f_{\tau_3}(x_3), ..., f_{\tau_{n+1}}(x_{n+1}))) \\ &= B(f_{\tau_1}(x_1), ..., f_{\tau_n}(x_{n+1})),\ \text{for appropriate } \tau\ \square \end{aligned}$$

The following lemma is analog to Lemma 16, for the R-calculus.

Lemma 28. If $\{f_\sigma\}$ is a partial homomorphism from M onto N, and g is an M-assignment, h is an N-assignment, $f_\sigma(g(x)) = h(x)$ for variables x of type σ, then $f_\sigma(\mathrm{Val}_1(s, g)) = \mathrm{Val}_2(s, h)$, for R-terms s of type σ.

Proof: By induction on s, where $M, N, \{f_\sigma\}$ are fixed. The variable, application, and λ-abstraction cases of the induction are as in the proof of Lemma 15.

We have $f_0(\mathrm{Val}_1(0, g)) = f_0(0) = 0 = \mathrm{Val}_2(0, h)$.

We must show that $f_{00}(\mathrm{Val}_1(N, g)) = \mathrm{Val}_2(N, h)$.

It suffices to show that for all $y \in \omega$, $f_0(A(\mathrm{Val}_1(N, g), y)) = B(\mathrm{Val}_2(N, h), f_0(y))$. Since f_0 is the identity, we have

$$\begin{aligned} f_0(A(\mathrm{Val}_1(N, g), y)) &= A(\mathrm{Val}_1(N, g), y) \\ &= y + 1 \\ &= B(\mathrm{Val}_2(N, h), y) \\ &= B(\mathrm{Val}_2(N, h), f_0(y)) \end{aligned}$$

Finally we must show that $f_{\tau\tau}(\mathrm{Val}_1(R_\sigma, g)) = \mathrm{Val}_2(R_\sigma, h)$, where $\tau = \sigma \to (0 \to \sigma)$. It suffices to show that $f_\tau(A(\mathrm{Val}_1(R\sigma, g), y)) = B(\mathrm{Val}_2(R_\sigma, h), f_\tau(y))$, for all $y \in \mathrm{dom}(f_\tau)$. It suffices to show $f_{0\sigma}(A(\mathrm{Val}_1(R_\sigma, g), y, z)) = B(\mathrm{Val}_2(R_\sigma, h), f_\tau(y), f_\sigma(z))$, for all $y \in \mathrm{dom}(f_\tau), z \in \mathrm{dom}(f_\sigma)$. Again it suffices to show $f\sigma(A(\mathrm{Val}_1(R_\sigma, y), y, z, k)) = B(\mathrm{Val}_2(R_\sigma, h), f_\tau(y), f_\sigma(z), k)$, for all $y \in \mathrm{dom}(f_\sigma), k \in \omega$. We show that this is true by induction on k. Note that $f_\sigma(A(\mathrm{Val}_1(R_\sigma, g), y, z, 0)) = f_\sigma(z) = B(\mathrm{Val}_2(R_\sigma, h), f_\tau(y), f_\sigma(z), 0)$. Assume true for k, and write

$$\begin{aligned} f_\sigma(A(\mathrm{Val}_1(R_\sigma, g), y, z, k_1)) &= f_\sigma(a(y, A(\mathrm{Val}_1(R_\sigma, g), y, z, k), k)) \\ &= B(f_\tau(y), f_\sigma(A(\mathrm{Val}_1(R_\sigma, g), y, z, k)), k) \\ &= B(f_\tau(y), B(\mathrm{Val}_2(R_\sigma, h), f_\tau(y), f_\sigma(z), k), k \\ &= B(\mathrm{Val}_2(R_\sigma, h), f_\tau(y), f_\sigma(z), k_1) \end{aligned}$$

by Lemma 27, since $y \in \mathrm{dom}(f_\tau)$, $A(\mathrm{Val}_1(R_\sigma, g), y, z, k) \in \mathrm{dom}(f_\sigma)$, $\ k \in \mathrm{dom}(f_0)$ □

Lemma 29. Suppose there is a partial homomorphism from M onto N, where M, N are R-structures. Then $M \vDash s = t$ implies $N \vDash s = t$, for any R-terms s, t.

Proof: Analogous to Lemma 16, using Lemma 29. □

Lemma 30. Let M be any R-structure. Then there is a partial homomorphism from T_ω onto M.

Proof: A special case of (the proof of) Lemma 18. □

Lemma 31. The relation $T_\omega \vDash s = t$ is Π^1_1.

Proof: We claim that $T_\omega \vDash s = t$ if and only if for all countable R-structures M, $M \vDash s = t$.

Suppose $T_\omega \vDash s = t$. Then by Lemma 30, all R-structures M have $M \vDash s = t$.

Suppose not $T_\omega \vDash s = t$. Assume $T_\omega \vDash s \neq t[f]$. Then let M be a countable elementary substructure of T_ω containing $\mathrm{Rng}(f)$, in the appropriate sense. M will be a countable R-structure, and $M \vDash s \neq t[f]$. So not $M \vDash s = t$.

We now wish to complete the proof the the relation $T_\omega \vDash s = t$ is complete Π^1_1.[2] To this end, let P be the set of indices of primitive recursive well orderings whose field is ω, whose least element is 0, and whose greatest element is 1. We can arrange the indexing so that every e is the index of a primitive recursive linear ordering $<_e$, whose field is ω, whose least element is 0, and whose greatest element is 1, and so that P is complete Π^1_1. □

Let $(a_0, ..., a_n, \overline{0})$ be the function f given by $f(i) = a_i$ for $i \leq n$, 0 otherwise.

Let $F : \omega^\omega \to \omega$. We wish to define $f = \Phi_e(F) \in \omega^\omega$. Let $f(0) = 1$. Let $f(n_1) = F((f(0), ..., f(n), \overline{0}))$ if $F((f(0), ..., f(n), \overline{0})) <_e f(n)$, 0 otherwise. Let $\Phi^*_e(F) = g$ be given by $g(n) = f(n)$ if $f(n+1) \neq 0$, 0 otherwise.

Lemma 32. If $e \in P$ then for all F, $F(\Phi^*_e(F)) = F(\overline{0})$ or $F(\Phi^*_e(F)) \neq 0$.

Proof: Since $e \in P$, let $\Phi_e(F)$ be $(a_0, ..., a_n, \overline{0})$, where $n \geq 0$, $a_0 = 1$, and $a_0 >_e ... >_e a_n$, $a_n \neq 0$. If $n = 0$ then $\Phi^*_e(F) = \overline{0}$, and we are done. Otherwise $\Phi^*_e(F) = (a_0, ..., a_{n-1}, \overline{0})$. Hence $a_n = F(\Phi^*_e(F))$. So $F(\Phi^*_e(F)) \neq 0$. □

[2] This half of the proof was motivated by a proof by R. Gandy and G. Kreisel (Communicated to us by H. Barendregt) which showed that there are two unequal p.r. functionals which agree on all primitive recursive functional arguments.

Lemma 33. For all e, $e \in P$ if and only if for all F, $F(\Phi_e^*(F)) = F(\overline{0})$ or $F(\Phi_e^*(F)) \neq 0$.

Proof: Assume $e \notin P$. Let $(a_0, a_1, a_2, ...)$ be such that $a_{i+1} <_e a_i$, and $a_0 = 1$. Clearly each $a_i \neq 0$, and so we may choose $F : \omega^\omega \to \omega$ such that $F(\overline{0}) = 1$, $F((a_0, a_1, a_2, ...)) = 0$, and $F((a_0, ..., a_n, \overline{0}) = a_{n+1}$ for $0 \leq n$. Clearly $\Phi_e(F) = \Phi_e^*(F) = (a_0, a_1, a_2, ...)$. Hence $F(\Phi_e^*(F)) = 0 \neq F(\overline{0})$. □

For each e, we let ψ_e be the functional of type $((0 \to 0) \to 0) \to 0$ given by $\psi_e(F) = 0$ if $F(\Phi_e^*(F)) = F(\overline{0})$ or $F(\Phi_e^*(F)) \neq 0$, 1 otherwise.

Lemma 34. For all e, $e \in P$ if and only if ψ_e is constantly 0.

Proof: Obvious from Lemma 33. □

Lemma 35. There is a total recursive function α such that for each e, $\alpha(e)$ is a closed R-term of type $((0 \to 0) \to 0) \to 0$ such that in T_ω, Val$(\alpha(e)) = \psi_e$.

Proof: This just says that ψ_e is a primitive recursive functional, defined effectively from e. □

Theorem 6. The relation $T_\omega \vDash s = t$, for R-terms s, t, is complete Π_1^1.

Proof: By Lemma 31, the relation is Π_1^1. By Lemmas 34 and 35, together with the fact that P is complete Π_1^1, we see that the relation is complete Π_1^1. □

References

[1] Luis Elpidio Sanchis et al. "Functionals defined by recursion." In: *Notre Dame Journal of Formal Logic* 8.3 (1967), pp. 161–174 (cit. on p. 7).

[2] William W Tait. "Intensional interpretations of functionals of finite type I". In: *The journal of symbolic logic* 32.2 (1967), pp. 198–212 (cit. on p. 7).

[3] Hendrik Pieter Barendregt. "Some extensional term models for combinatory logics and lambda-calculi". PhD thesis. Utrecht: Rijksuniversiteit Utrecht, 1971 (cit. on p. 7).