On Undefined and Meaningless in Lambda Definability*

Fer-Jan de Vries

Computer Science, University of Leicester, Leicester, UK
fdv1@le.ac.uk

Abstract

We distinguish between undefined terms as used in lambda definability of partial recursive functions and meaningless terms as used in infinite lambda calculus for the infinitary terms models that generalise the Böhm model. While there are uncountable many known sets of meaningless terms, there are four known sets of undefined terms. Two of these four are sets of meaningless terms.

In this paper we first present set of sufficient conditions for a set of lambda terms to serve as set of undefined terms in lambda definability of partial functions. The four known sets of undefined terms satisfy these conditions.

Next we locate the smallest set of meaningless terms satisfying these conditions. This set sits very low in the lattice of all sets of meaningless terms. Any larger set of meaningless terms than this smallest set is a set of undefined terms. Thus we find uncountably many new sets of undefined terms.

As an unexpected bonus of our careful analysis of lambda definability we obtain a natural modification, strict lambda-definability, which allows for a Barendregt style of proof in which the representation of composition is truly the composition of representations.

1998 ACM Subject Classification F.4.1 [Mathematical Logic] lambda calculus and related systems

Keywords and phrases lambda calculus, lambda definability, partial recursive function, undefined term, meaningless term

Digital Object Identifier 10.4230/LIPIcs.FSCD.2016.18

1 Introduction

The intuition that not all lambda terms are equally *significant* from a computational point of view is as old as lambda calculus itself. It is of particular interest that in lambda calculus, unlike e.g. the notion of zero in arithmetic, the notion of insignificant term is not uniquely determined. There are many different reasonable choices that one can make for a set of unsignificant terms. Making a concrete choice is akin to choosing a semantics for the lambda calculus.

The oldest, relatively understudied application of insignificant terms is made in the lambda definability of partial recursive functions. In this area insignificant terms are traditionally called *undefined terms*. The other more modern and better understood application is the construction of infinitary term models of the lambda calculus. This construction generalises the Böhm model. In the latter case the insignificant terms are called *meaningless terms*.

* Dedicated in friendship to Albert Visser on the occasion of his retirement.

1st International Conference on Formal Structures for Computation and Deduction (FSCD 2016).
Editors: Delia Kesner and Brigitte Pientka; Article No. 18; pp. 18:1–18:15
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

There are four well-known sets of undefined terms used in lambda definability. In contrast there are uncountably many sets of meaningless terms, that each give rise to their own model of lambda calculus. As it happens only two of the four known sets of undefined terms are also sets of meaningless terms. Hence it is a natural question to ask which of the uncountable many other sets of meaningless terms can also play the role of set of undefined terms, so that in the corresponding model the recursive functions can naturally be interpreted.

The proof technique of Statman's theorem which, as Barendregt has shown, works uniformly for each of the four known sets of undefined terms does not generalise to arbitrary sets of meaningless terms, because in general sets of meaningless terms are not co-Visser-sets. Instead we analyse the proof of lambda definability in Barendregt's PhD thesis. With our modern insight in sets of meaningless terms we can generalise and improve this old proof, but we can also improve it.

Church and Kleene were the first to give lambda-representation of recursive functions. In their representation the role of undefined terms is played by the terms without a finite normal form. Barendregt criticises their representation and expresses the clear ideal that the lambda-representation of recursive functions should preserve the way they are defined (this wish is also known as Kreisel's Superthesis). The Church-Kleene lambda-representation falls short of this ideal: the representation of the composition of two recursive functions is not the composition of their represenation. Barendregt then takes the rather revolutionary step to replace their old notion of undefined term by the new concept of unsolvable term.[1]

This new notion of undefined term indeed allows for an improvement of the old proof. But there is a surprise. Barendregts lambda-represenation of partial recursive functions falls arguably short of his own ideal. He gives first a lambda-representation of the total recursive functions and then a lambda representation of the partial recursive functions. In case of the total functions he use the natural notion of composition of their representations. In case of the partial recursive functions he defines composition in a slightly ad hoc way. He gets around this definition by using a very clever "jamming" trick.

We observe in this paper that by using the novel concept of strict lambda definability we can represent the partial recursive functions in such a way that their definition is completely preserved, in line with Barendregt's original "dream improvement" of the old proof by Church. This reformulated proof is also more general: it now applies to any set of meaningless terms satisfying one particular extra closure condition.

Ordered by inclusion the sets of meaningless terms form a lattice. The largest set of meaningless terms that can be used to make a model of lambda calculus is the set of unsolvables. The smallest such set is the set of rootactive terms[2]. The infinitary term models constructed with these two sets of meaningless terms are the Böhm model and the Berarducci model. In the Böhm model they are exactly the unsolvables that are equated with $\perp$. In the Berarducci model they are the rootactives that are equated with $\perp$.

The smallest set of undefined terms that we can identify sits very low in this lattice of sets of meaningless terms just above the the set of rootactives. This raises an open question: whether the partial recursive functions can be interpreted in the Berarducci model of the

---

[1] A closed lambda term M is *solvable* if $MN_1 \dots N_k = \mathbf{I}$ for some sequence $N_1, \dots, N_k$ with $k \geq 0$. An open lambda term is called *solvable* if its closure is solvable. A lambda terms is called *unsolvable* if it is not solvable.

[2] A lambda term M is rootactive if any reduct of M can further reduce to a redex. The classical rootactive term is $\mathbf{\Omega}$. The unsolvable $\Omega\mathbf{I}$ is not rootactive. Note that the definition of a rootactive terms allows for free variables. The term EyE with $E \equiv \mathbf{\Theta}\lambda xyz.zyx$ is a concrete example of a rootactive term with free variable y.

lambda calculus in such a way that if a partial recursive function f is not defined on a natural number n, then the interpretation of $f(n)$ equals bottom.

2 Brief recap of 80 years of lambda definability

Lambda definability goes some 80 years back to the exciting, early days of lambda calculus when, in the slipstream of Gödel's incompleteness theorems, Church and his students Kleene and Rosser were experimenting which functions could be represented in the lambda calculus, Gödel and Herbrand defined the recursive functions and Turing tried to capture the intuitive idea of "effective calculable" function with his machines. While Church was mainly using the $\lambda\mathbf{I}$-calculus[3] considers only in his papers, Turing realised that it is "naturally much simpler" [22] to use $\lambda\mathbf{K}$-calculus, what we now call the lambda calculus.

That recursive functions on natural numbers can be represented in lambda calculus is due to Kleene [15]. The converse, that lambda-definable functions on natural numbers are recursive, is due to Kleene and Church independently [15, 9]. These results are important as on one hand they led Church to his Church Thesis and on the other hand they demonstrate that lambda calculus is a paradigmatic programming language [3].

The first definition of lambda definability dealt with total functions, as partial recursive functions had not yet been defined.

▶ **Definition 1.** A total function $f : \mathbb{N} \to \mathbb{N}$ is λ-*definable* if for some lambda term F and each $n \in \mathbb{N}$ we have $F\ulcorner n\urcorner = \ulcorner f(n)\urcorner$.

▶ **Theorem 2.** *A total function $f : \mathbb{N} \to \mathbb{N}$ is λ-definable if and only if f is recursive.*

When Kleene [16] defined next the partial recursive functions, Theorem 2 was immediately extended to partial recursive functions [10, 11]. This required an extra clause to Definition 1 to explain what happens when the function that one wants to represent happens to be undefined on some input.

▶ **Definition 3.** Let $\mathcal{U}$ be a set of lambda terms. A partial function $\phi : \mathbb{N} \nrightarrow \mathbb{N}$ is $\lambda_{\mathcal{U}}$-*definable* if for some lambda term F and each $n \in \mathbb{N}$:

$$\begin{array}{ll} F\ulcorner n\urcorner = \ulcorner \phi(n)\urcorner & \text{if } \phi(n)\downarrow \\ F\ulcorner n\urcorner \in \mathcal{U} & \text{else.} \end{array}$$

2.1 Kleene-Church: undefined is having no normal form

Church [10, p. 29] used the set $\overline{\mathcal{NF}}$ lambda terms without normal form for $\mathcal{U}$ to represent "undefined".

▶ **Theorem 4** (Kleene). *A partial function $\phi : \mathbb{N} \nrightarrow \mathbb{N}$ is $\lambda_{\overline{\mathcal{NF}}}$-definable if and only if ϕ is partial recursive.*

In his thesis [1, 3] Barendregt points out that there is a practical problem with composition in the approach of Church and Kleene. If f, g are $\lambda_{\overline{\mathcal{NF}}}$-defined by F, G, then $f \circ g$ can not be $\lambda_{\overline{\mathcal{NF}}}$-defined by $\lambda x.F(Gx)$, which would be the natural way to $\lambda_{\overline{\mathcal{NF}}}$-define $F \circ G$. As example, Barendregt takes for f the constant zero function represented by $\lambda x.\ulcorner 0\urcorner$ and for g the function that is everywhere undefined represented by $\mathbf{\Omega}$. Then $f \circ g$ is totally undefined,

[3] The $\lambda\mathbf{I}$-calculus allows terms of the form $\lambda x.M$ only if x is a free variable of M.

but $F \circ G \equiv \lambda x.F(Gx) = \lambda x.\ulcorner 0 \urcorner$. The conclusion is then that "it is not immediate that the $\lambda_{\overline{\mathcal{NF}}}$-definable functions are closed under composition." Kleene and Church avoid this issue by using Kleene's normal form theorem. They represent only the normal form of a partial recursive function. Barendregt emphasises that their representation of the partial recursive functions is not intensional, as it does not preserve their definition trees.

2.2 Barendregt: undefined is being unsolvable

Barendregt's solution is to take for $\mathcal{U}$ the set $\overline{\mathcal{HNF}}$[4] of unsolvables.

▶ **Theorem 5** (Barendregt). *A partial function $\phi : \mathbb{N} \rightarrow\!\!\!\!\!+ \mathbb{N}$ is $\lambda_{\overline{\mathcal{HNF}}}$-definable if and only if ϕ is partial recursive.*

Barendregt uses Lercher's jamming factor trick and represents the composition $f \circ g$ by $\lambda x.Gx\mathbf{KII}F(Gx)$. This clever trick works because if $g(n)$ is undefined then $G\ulcorner n \urcorner$ is unsolvable and hence also $F\ulcorner n \urcorner \equiv G\ulcorner n \urcorner\mathbf{KII}F(G\ulcorner n \urcorner)$ is unsolvable, and if $g(n)$ is defined then it can be shown that $G\ulcorner n \urcorner\mathbf{KII} \twoheadrightarrow_\beta \mathbf{I}$ in which case $F\ulcorner n \urcorner = F(G\ulcorner n \urcorner)$. Cf. [3, Lemma 8.4.5].

Barendregt felt strongly about this change from $\overline{\mathcal{NF}}$ to $\overline{\mathcal{HNF}}$ in the definition of lambda definability. In his thesis he writes on page xvi: "This is not to be regarded as a mere technical improvement but simply central to the objects which are here intended." And in [4] he explains in detail:

> It has been stressed by Kreisel [18, p. 177-178] that in connection with the so-called "superthesis", Church's thesis expresses less than we know. When we say that all mechanically computable number theoretic functions are λ-definable or recursive, we merely speak of the results of computations, of their graphs. But we have in mind that λ-terms correspond to our procedures for defining these functions. As far as the μ-recursive functions and the λ-definable functions are concerned, strong definability proves the equivalence not only in the sense of Church but also of the super thesis: definitions are preserved. [4]

Given these strong arguments against the traditional Kleene-Church proof of $\lambda_{\overline{\mathcal{NF}}}$-definability, it is a bit unexpected that composition is not defined in the natural way as $\lambda x.F(Gx)$ in $\lambda_{\overline{\mathcal{HNF}}}$-definability. Barendregt's solution to deal with composition is arguably almost but not quite in the spirit of the superthesis.

In Section 3 we will show that Lercher's jamming factor trick is not needed for partial functions either, and that composition can indeed be represented compositionally.

2.3 Statman: undefined is belonging to a co-Visser set

Statman takes a general approach: any non-empty co-Visser set of closed lambda terms can be used as set $\mathcal{U}$ of undefined terms in $\lambda_\mathcal{U}$-definability of the partial recursive functions.

▶ **Definition 6.** A set $\mathcal{U} \subseteq \Lambda^0$ is a co-Visser set, if:

1. $\Lambda^0 \backslash \mathcal{U}$ is recursive enumerable,
2. $\Lambda^0 \backslash \mathcal{U}$ is closed under finite β-reduction.

▶ **Theorem 7** (Statman, 1990). *Let $\mathcal{A}$ be a non-empty co-Visser set. Then any partial recursive function is $\lambda_\mathcal{A}$-definable.*

4 Wadsworth has shown that a term is unsolvable iff it has no head normal form. Hence our notation $\overline{\mathcal{HNF}}$ for the set of unsolvables.

Barendregt [5] has given a detailed proof of Statman's Theorem and Visser's Anti Diagonalisation Theorem [24, Thm. 4.4] on which Statman's Theorem is based. This way of proving lambda definability has two consequences. First, Statman's theorem is formulated for closed terms. This is because the proof makes use of a self-interpreter, i.e. a lambda term $\mathbf{E}$ such that for $M \in \Lambda^0$ one has

$$\mathbf{E}\ulcorner \# M \urcorner \twoheadrightarrow_\beta M,$$

where $\# : \Lambda \to \mathsf{Nat}$ is some effective bijection that assigns to a lambda term a unique natural number. This equation cannot hold when M contains free variables [3, Definition 8.5.1]. Secondly, the Statman proof does not give support for Kreisel's superthesis. Visser's theorem uses Ershov's precomplete numerations, so that the proof of Visser's theorem is '"coordinate free" i.e. the proof uses (nearly) no specific properties of lambda calculus' in the words of [24].

Barendregt lists four sets that satisfy the condition of Statman's theorem: (the subsets of closed terms of) $\overline{\mathcal{NF}}$, $\overline{\mathcal{HNF}}$, the set $\overline{\mathcal{WHNF}}$ of terms without a weak head normal form[5] and the set $\mathcal{E}$ of easy[6] terms. Two of these, $\overline{\mathcal{HNF}}, \overline{\mathcal{WHNF}}$ are sets of meaningless terms. We will see in Section 6 that there are many other sets of meaningless terms which don't satisfy the Statman condition and yet can be used as set of undefined terms.

3 Strict $\lambda_{\mathcal{U}}$-definability

In this section we search for general sufficient conditions for a set $\mathcal{U}$ of lambda terms so that we can generalise Barendregt's proof of lambda definability. We follow the notation of [3] for the standard Church coding:

$$\begin{array}{llllll}
\mathbf{F} & \equiv & \lambda xy.y & \mathbf{T} & \equiv & \lambda xy.x \\
\mathbf{and} & \equiv & \lambda xy.xyx & \mathbf{if}\ B\ \mathbf{then}\ M\ \mathbf{else}\ N & \equiv & BMN \\
[M,N] & \equiv & \lambda z.zMN & \mathbf{Zero} & \equiv & \lambda x.x\mathbf{T} \\
\mathbf{S}^+ & \equiv & \lambda x.[\mathbf{T},x] & \mathbf{P}^- & \equiv & \lambda x.[\mathbf{F},x] \\
\mathbf{K} & \equiv & \lambda xy.x & \mathbf{I} & \equiv & \lambda x.x
\end{array}$$

▶ **Definition 8** (Turing's fixed point combinator [23]). We define:

$$\mathbf{\Theta} \equiv (\lambda xy(y(xxy))\lambda xy(y(xxy)).$$

▶ **Definition 9** (Barendregt numerals [2]). We define $\ulcorner\ \urcorner : \mathbb{N} \to \Lambda$ by induction:

$$\begin{array}{lll}
\ulcorner 0 \urcorner & \equiv & \mathbf{I} \\
\ulcorner n+1 \urcorner & \equiv & [\mathbf{F}, \ulcorner n \urcorner]
\end{array}$$

Let us first follow [3] and define the class of partial recursive functions as the least class of partial numeric functions which contains the total recursive functions and is closed under composition and minimalisation.

Suppose our candidate set of undefined terms is $\mathcal{U}$. We will inspect Barendregt's proof to see what requirements we have to make on $\mathcal{U}$.

---

[5] A lambda term *has a weak head normal form* if it can reduce to either an abstraction or a term of the form $xM_1 \ldots M_n$.

[6] A lambda term is *easy* if it can consistently be equated to any other lambda term.

3.1 Composition

Let F, G be the representations of the partial numeric functions $f, g : \mathbb{N} \to \mathbb{N}$. The natural way to represent composition is

$$(F \circ G) \equiv (\lambda x.F(Gx)),$$

as in the lambda definability proof of the total recursive functions in [3]. This works all right in case g is well-defined on n and f is defined on $g(n)$, because then

$$\ulcorner f \circ g \urcorner \ulcorner n \urcorner \equiv (F \circ G)\ulcorner n \urcorner \twoheadrightarrow_\beta F(G\ulcorner n \urcorner) \equiv F\ulcorner g(n) \urcorner \equiv \ulcorner f(g(n)) \urcorner.$$

If $g(n)$ is undefined, then $G\ulcorner n \urcorner$ should reduce to some $U \in \mathcal{U}$. With the notion of $\lambda_\mathcal{U}$ definability we cannot infer from $F(G\ulcorner n \urcorner) \twoheadrightarrow_\beta FU$ that $FU \in \mathcal{U}$. Barendregt's jamming trick argument can be repeated, provided that the set $\mathcal{U}$ of undefined terms has the property: if $U \in \mathcal{U}$ then $UM \in \mathcal{U}$ for any $M \in \Lambda$. In particular the set of unsolvables has this property.

However, the jamming trick and the previous condition on $\mathcal{U}$ is not needed with the following "stricter" definition of lambda definability:

▶ **Definition 10.** Let $\mathcal{U}$ be a set of lambda terms. A partial function $\phi : \mathbb{N}^p \rightharpoonup \mathbb{N}$ is *strictly $\lambda_\mathcal{U}$-definable* if for some lambda term F and each $\vec{n} \in \mathbb{N}^p$

1. $F\ulcorner \vec{n} \urcorner \twoheadrightarrow \ulcorner m \urcorner$ if $\phi(\vec{n}) = m$,
2. $F\ulcorner \vec{n} \urcorner \in \mathcal{U}$ if $\phi(\vec{n}) \uparrow$,
3. $F\vec{N} \in \mathcal{U}$ for all $\vec{N} \in \Lambda^p$ with at least one $N_i \in \mathcal{U}$.

This new strictness clause does the trick for proving that the representation of composition is the composition of the representations:

▶ **Lemma 11.** *The strictly $\lambda_\mathcal{U}$-definable partial functions are closed under composition.*

Proof. To keep the notation simple we consider without loss of generality unary numeric functions. Let F, G be the strict representations of the partial numeric functions f, g. Then for $U \in \mathcal{U}$ we have $GU \in \mathcal{U}$ and hence also

$$(F \circ G)U \equiv (\lambda x.F(Gx))U \to F(GU) \in \mathcal{U}.$$

For $n \in \mathbb{N}$ we have either $g(n) \downarrow$ or $g(n) \uparrow$. If the former than

$$(F \circ G)\ulcorner n \urcorner \equiv (\lambda x.F(Gx))\ulcorner n \urcorner \to F(G\ulcorner n \urcorner) \twoheadrightarrow F\ulcorner g(n) \urcorner \twoheadrightarrow \ulcorner f(g(n)) \urcorner.$$

If the latter, then we have that $G\ulcorner n \urcorner \twoheadrightarrow_\beta U$ for some $U \in \mathcal{U}$ and therefore by the new strictness clause we get

$$(F \circ G)\ulcorner n \urcorner \equiv (\lambda x.F(Gx))\ulcorner n \urcorner \to F(G\ulcorner n \urcorner) \twoheadrightarrow FU \in \mathcal{U}.$$ ◀

3.2 Minimalisation

In [3] a lambda term P is called a predicate if $P\ulcorner n \urcorner$ reduces to either $\mathbf{T}$ or $\mathbf{F}$ for all $n \in \mathbb{N}$. We will use in this section *strict* predicates that satisfy the extra property that $PU \in \mathcal{U}$ whenever $U \in \mathcal{U}$ for some fixed set $\mathcal{U}$ of undefined terms.

In the proof of [3, Prop. 8.4.10] we find this definition of a lambda term:

$$H_P \equiv \mathbf{\Theta}(\lambda hx.\mathbf{if}\ Px\ \mathbf{then}\ x\ \mathbf{else}\ h(\mathbf{S}^+x))$$

where P is a predicate, together with the following reduction

$$H_P\ulcorner n\urcorner \twoheadrightarrow_\beta \textbf{if } P\ulcorner n\urcorner \textbf{ then } \ulcorner n\urcorner \textbf{ else } H_P\ulcorner n+1\urcorner.$$

Hence, this finite term H_P can reduce with an infinite reduction to the infinite expression

$$\textbf{if } P\ulcorner 0\urcorner \textbf{ then } \ulcorner 0\urcorner \textbf{ else if } P\ulcorner 1\urcorner \textbf{ then } \ulcorner 1\urcorner \textbf{ else if } P\ulcorner 2\urcorner \textbf{ then } \ulcorner 2\urcorner \textbf{ else } \dots$$

Recall that minimalisation is the construction of a new partial function

$$\mu m[\chi(\vec{n}, m) = 0] : \mathbb{N}^p \to \mathbb{N}$$

from a given partial function $\chi : \mathbb{N}^{p+1} \to \mathbb{N}$. The new function calculates for given $\vec{n}$ the least m such that $\chi(\vec{n}, m) = 0$, if there is such an m and is undefined otherwise.

Suppose χ is $\lambda_{\mathcal{U}}$-defined by G. We will now represent $\mu m[\chi(\vec{n}, m) = 0]$ by the lambda term $\lambda\vec{n}.((\lambda p.H_p\ulcorner 0\urcorner)\ \lambda m.\mathbf{Zero}(G\vec{n}m))$. we can safely say that this representation preserves the definition of minimalisation. After all, instead of the notation $\mu m[\chi(\vec{n}, m) = 0]$ one could just as well have opted for $\mu[\lambda m.\chi(\vec{n}, m) = 0]$ instead.

In [3] we find the following proposition:

▶ **Proposition 12** ([3, Prop. 8.4.10]). *Let P be such that for all $n \in \mathbb{N}$ one has $P\ulcorner n\urcorner \twoheadrightarrow \mathbf{F}$. Then:*

1. *μP has no normal form,*
2. *μP is unsolvable.*

Since we want to generalise from the set of unsolvables to other sets of undefined terms, we can not use the previous proposition. However, we can reuse its proof, which actually shows that μP is rootactive.[7]

This leads us to a more general proposition:

▶ **Proposition 13.** *Let P be such that for all $n \in \mathbb{N}$ we have $P\ulcorner n\urcorner \twoheadrightarrow \mathbf{F}$. Then$\mu P$ is rootactive.*

Proof. Consider the following reduction from [3] that we reproduce here with slightly more detail:

$$\begin{array}{lll}
\mu P & \equiv & H_P\ulcorner 0\urcorner \\
 & \twoheadrightarrow & \textbf{if } P\ulcorner 0\urcorner \textbf{ then } \ulcorner 0\urcorner \textbf{ else } H_P\ulcorner 1\urcorner \\
 & \twoheadrightarrow & \textbf{if } \mathbf{F} \textbf{ then } \ulcorner 0\urcorner \textbf{ else } H_P\ulcorner 1\urcorner \\
 & \equiv & \mathbf{F}\ulcorner 0\urcorner(H_P\ulcorner 1\urcorner) \\
 & \equiv & (\lambda xy.y)\ulcorner 0\urcorner(H_P\ulcorner 1\urcorner) \\
 & \to & (\lambda y.y)(H_P\ulcorner 1\urcorner) \\
 & \to_0 & H_P\ulcorner 1\urcorner \\
 & \twoheadrightarrow & \textbf{if } \mathbf{F} \textbf{ then } \ulcorner 1\urcorner \textbf{ else } H_P\ulcorner 2\urcorner \\
 & \twoheadrightarrow & \lambda y.y(H_P\ulcorner 2\urcorner) \\
 & \to_0 & H_P\ulcorner 2\urcorner \\
 & \twoheadrightarrow & \dots
\end{array}$$

In all segments $H_P\ulcorner n\urcorner \twoheadrightarrow H_P\ulcorner n+1\urcorner$ at least one reduction step takes place at the root (the step $\to_0$). Hence using the terminology of [13] the infinite reduction starting from μP

[7] It is well known that the set of rootactives is a proper subset of the set of unsolvables. E.g. $\lambda x.\Omega$, $\lambda x.\Omega x$ and $\mathbf{\Theta K}$ are unsolvables that are not rootactive.

is hypercollapsing.[8] Hence by [13, Theorem 12.8.3] we obtain that the initial term μP is rootactive. ◀

▶ **Lemma 14.** *Let $\mathcal{U}$ be a set of lambda terms such that $\mathcal{U}$ contains all rootactive terms and $U \in \mathcal{U}$ implies $U\mathbf{T}\ulcorner n \urcorner M \in \mathcal{U}$, for any $M \in \Lambda$. The strictly $\lambda_{\mathcal{U}}$-definable partial functions are closed under minimalisation.*

Proof. Let $\mathcal{U}$ be a set of terms such that $\mathcal{U}$ contains all rootactive terms and $U \in \mathcal{U}$ implies $U\mathbf{T}\ulcorner n \urcorner M \in \mathcal{U}$. Let $\phi(\vec{n}) \equiv \mu m[\chi(\vec{n}, m) = 0]$, where χ is total and $\lambda_{\mathcal{U}}$-definable by, say, G. Then we define:

$$F \equiv \lambda\vec{x}.\mu[\lambda y.\mathbf{Zero}(G\vec{x}y)].$$

If $\phi(\vec{n})\downarrow$, then $\chi(\vec{n}, m) = 0$ for some $m \in \mathbb{N}$. Then by [3, Lemma 6.3.9(ii)] we get

$$F\ulcorner\vec{n}\urcorner = \ulcorner\phi(\vec{n})\urcorner.$$

And if $\phi(\vec{n})\uparrow$, then $\chi(\vec{n}, m) \neq 0$ and so $\mathbf{Zero}(G\vec{n}m) \twoheadrightarrow_\beta \mathbf{F}$ for all $m \in \mathbb{N}$. Hence by Proposition 13 we see that

$$F\ulcorner\vec{n}\urcorner \twoheadrightarrow_\beta \mu[\lambda y.\mathbf{Zero}(G\vec{n}y)]$$

is rootactive. Finally, consider an $\vec{N} \in \Lambda$ with at least one $N_i \in \mathcal{U}$. Because G is strict, this implies the existence of a $U \in \mathcal{U}$ such that $G\vec{N}\ulcorner 0 \urcorner \twoheadrightarrow_\beta U$. We can now make the following reduction:

$$\begin{array}{rcl} F\vec{N} & \twoheadrightarrow_\beta & \mu[\lambda y.\mathbf{Zero}(G\vec{N}y)] \\ & \twoheadrightarrow_\beta & H_P\ulcorner 0\urcorner \\ & \twoheadrightarrow_\beta & \mathbf{if}\ P\ulcorner 0\urcorner\ \mathbf{then}\ \ulcorner 0\urcorner\ \mathbf{else}\ H_P\ulcorner 1\urcorner \\ & \twoheadrightarrow_\beta & \mathbf{if}\ \mathbf{Zero}(G\vec{N}\ulcorner 0\urcorner)\ \mathbf{then}\ \ulcorner 0\urcorner\ \mathbf{else}\ H_P\ulcorner 1\urcorner \\ & \twoheadrightarrow_\beta & \mathbf{if}\ \mathbf{Zero}\ U\ \mathbf{then}\ \ulcorner 0\urcorner\ \mathbf{else}\ H_P\ulcorner 1\urcorner \\ & \equiv & \mathbf{Zero}\ U\ \ulcorner 0\urcorner\ (H_P\ulcorner 1\urcorner) \\ & \equiv & (\lambda x.x\mathbf{T})U\ulcorner 0\urcorner(H_P\ulcorner 1\urcorner) \\ & \twoheadrightarrow_\beta & U\mathbf{T}\ulcorner 0\urcorner(H_P\ulcorner 1\urcorner) \end{array}$$

where P stands for $\lambda y.\mathbf{Zero}(G\vec{N}y)$. The last term $U\mathbf{T}\ulcorner 0\urcorner(H_P\ulcorner 1\urcorner)$ is undefined because of our assumption on $\mathcal{U}$.

Concluding, we have shown that $\phi(\vec{n})$ is strictly $\lambda_{\mathcal{U}}$-definable by F. ◀

3.3 Total recursive functions

After composition and minimalisation we will now look at a strict encoding of the total recursive functions. This presents another obstacle: the usual representation of a total recursive function is not strict. Consider for instance the constant 0 function represented by $\mathbf{Z} \equiv \lambda x.\ulcorner 0\urcorner$ in [3]. We get:

$$(\lambda x.\ulcorner 0\urcorner)\mathbf{\Omega} \equiv (\lambda x.\mathbf{I})\mathbf{\Omega} \to \mathbf{I}.$$

[8] A hypercollapsing reduction is a "quasi root reduction," i.e. a reduction containing infinitely many root reduction steps of the form $(\lambda x.M)N \to_0 M[x := N]$.

Our previous analysis of minimalisation forced upon us the condition for $\mathcal{U}$, that if $U \in \mathcal{U}$ then also $U\mathbf{T}XY \in \mathcal{U}$ for any $X, Y \in \Lambda$. Now, if $\mathbf{I}$ would be an element of $\mathcal{U}$, then for any $M \in \Lambda$ we would get:

$$\mathbf{IT}MM \equiv \mathbf{IK}MM \to_\beta \mathbf{K}MM \to_\beta M \in \mathcal{U}.$$

This can not be the case, as the numerals are not supposed to be undefined.

There is, however, a strict way of representing the constant zero function. Consider the following infinite expression:

$$\lambda x.\mathbf{if}\ x = \ulcorner 0 \urcorner\ \mathbf{then}\ \ulcorner 0 \urcorner\ \mathbf{else}\ \mathbf{if}\ x = \ulcorner 1 \urcorner\ \mathbf{then}\ \ulcorner 0 \urcorner\ \mathbf{else}\ \mathbf{if}\ x = \ulcorner 2 \urcorner\ \mathbf{then}\ \ulcorner 0 \urcorner\ \mathbf{else}\ \ldots$$

in which we use $x = \ulcorner m \urcorner$ as shorthand for $\mathbf{Zero}(\mathbf{P}^{-m}\ulcorner x \urcorner)$. Clearly, this expression will reduce to $\ulcorner 0 \urcorner$ whenever x is a numeral $\ulcorner n \urcorner$. The above infinite expression is of the simple form

$$\lambda x.\mathbf{if}\ x = \ulcorner 0 \urcorner\ \mathbf{then}\ X\ \mathbf{else}\ Y \text{ for some possibly infinite expressions } X, Y.$$

If we provide the previous term with input $U \in \mathcal{U}$, we get:

$$\begin{aligned} \mathbf{if}\ U = \ulcorner 0 \urcorner\ \mathbf{then}\ X\ \mathbf{else}\ Y' &\twoheadrightarrow_\beta \lambda x.\mathbf{if}\ \mathbf{Zero}\ U\ \mathbf{then}\ X\ \mathbf{else}\ Y \\ &\twoheadrightarrow_\beta \mathbf{Zero}\ UXY \\ &\equiv (\lambda x.x\mathbf{T})UXY \\ &\twoheadrightarrow_\beta U\mathbf{T}XY \end{aligned}$$

We find that expression $\lambda x.\mathbf{if}\ x = \ulcorner 0 \urcorner\ \mathbf{then}\ X\ \mathbf{else}\ Y$ is strict for those sets $\mathcal{U}$ that satisfy the same property that we needed in Lemma 14, namely that $U \in \mathcal{U}$ must imply $U\mathbf{T}\ulcorner 0 \urcorner M \in \mathcal{U}$ for any $M \in \Lambda$.

In general, where Barendregt would use F to represent a total unary function f, we transform his F to a finite term which can reduce to an infinite representation for f:

$$\lambda n.\mathbf{if}\ n = \ulcorner 0 \urcorner\ \mathbf{then}\ F\ulcorner 0 \urcorner\ \mathbf{else}\ \mathbf{if}\ n = \ulcorner 1 \urcorner\ \mathbf{then}\ F\ulcorner 1 \urcorner\ \mathbf{else}\ \mathbf{if}\ n = \ulcorner 2 \urcorner\ \mathbf{then}\ F\ulcorner 2 \urcorner\ \mathbf{else}\ \ldots.$$

This representation is strict, with a similar argument. And for all input of the form $\ulcorner n \urcorner$ with $n \in \mathbb{N}$ the expression reduces to $F\ulcorner n \urcorner$. Thanks to the fixed-point trickery of Turing [22] there is a finite term that reduces to this this infinite expression:

$$L \equiv \lambda f.\mathbf{\Theta}(\lambda wmn.(\mathbf{Zero}\ n)(fm)(w(\mathbf{P}^-\ n)(\mathbf{S}^+\ m)))$$

If F is the Church-Barendregt encoding of a total unary numeric function f, we get:

$$\begin{aligned} LF\ulcorner 0 \urcorner\ulcorner n \urcorner &\twoheadrightarrow \mathbf{if}\ \mathbf{Zero}\ulcorner n \urcorner\ \mathbf{then}\ F\ulcorner 0 \urcorner\ \mathbf{else}\ KF\ulcorner 1 \urcorner(\mathbf{P}^-\ulcorner n \urcorner) \\ &\twoheadrightarrow_\beta \mathbf{if}\ \mathbf{Zero}\ (\mathbf{P}^-\ulcorner n \urcorner)\ \mathbf{then}\ F\ulcorner 1 \urcorner\ \mathbf{else}\ LF\ulcorner 2 \urcorner(\mathbf{P}^{-2}\ulcorner n \urcorner) \\ &\twoheadrightarrow_\beta \ldots \\ &\twoheadrightarrow_\beta \mathbf{if}\ \mathbf{Zero}\ (\mathbf{P}^{-n}\ulcorner n \urcorner)\ \mathbf{then}\ F\ulcorner n \urcorner\ \mathbf{else}\ LF\ulcorner n+1 \urcorner(\mathbf{P}^{-n+1}\ulcorner n)\urcorner \\ &\twoheadrightarrow_\beta \mathbf{if}\ \mathbf{Zero}\ \ulcorner 0 \urcorner\ \mathbf{then}\ F\ulcorner n \urcorner\ \mathbf{else}\ LF\ulcorner n+1 \urcorner(\mathbf{P}^{-n+1}\ulcorner n)\urcorner \\ &\twoheadrightarrow_\beta F\ulcorner n \urcorner \end{aligned}$$

▶ **Lemma 15.** *Let $\mathcal{U}$ be a set of lambda terms such that $U \in \mathcal{U}$ implies $UXYZ \in \mathcal{U}$, for any $X, Y, Z \in \Lambda$. Then the total unary recursive functions are strictly $\lambda_{\mathcal{U}}$-definable.*

Proof. If F is the Barendregt representation of the total unary recursive function f, then we will represent f now by $\lambda x.LF\ulcorner 0 \urcorner x$. ◀

▶ **Corollary 16.** *The function* $S^+(n) = n + 1$ *is strictly* $\lambda_{\mathcal{U}}$*-definable by* $\lambda x.L\mathbf{S}^{+}\ulcorner 0 \urcorner x$ *and the function* $Z(n) = 0$ *is strictly* $\lambda_{\mathcal{U}}$*-definable by* $\lambda x.L\mathbf{Z}x$*, for any set* $\mathcal{U}$ *of lambda terms satisfying the condition* $U \in \mathcal{U}$ *implies* $U\mathbf{T}XY \in \mathcal{U}$*, for any* $X, Y \in \Lambda$*.*

Next we want to show that p-ary projections functions U_i^p can be strictly $\lambda_{\mathcal{U}}$-defined. With this goal in mind, consider the case where f is a constant function $\lambda n.g$ and g is total p-ary recursive function that can be $\lambda_{\mathcal{U}}$-defined by some lambda term G. Then we represent f by

$$\lambda n.L(\lambda m.G)\ulcorner 0 \urcorner n$$

which reduces to the infinite term

$$\lambda n.\mathbf{if}\ n = \ulcorner 0 \urcorner\ \mathbf{then}\ G\ \mathbf{else\ if}\ n = \ulcorner 1 \urcorner\ \mathbf{then}\ G\ \mathbf{else\ if}\ n = \ulcorner 2 \urcorner\ \mathbf{then}\ G\ \mathbf{else}\ \ldots.$$

▶ **Lemma 17.** *The functions* $U_i^p \equiv \lambda x_1 \ldots x_p.x_i$ *with* $0 \leq i \leq p$ *are strictly* $\lambda_{\mathcal{U}}$*-definable, for any set* $\mathcal{U}$ *of lambda terms satisfying the condition* $U \in \mathcal{U}$ *implies* $U\mathbf{T}XY \in \mathcal{U}$*, for any* $X, Y \in \Lambda$*.*

Proof Sketch. Note that U_i^p can be rewritten $\lambda \vec{x}.U_p^p \vec{y}$, where $\vec{y}$ is obtained from the sequence of variables $\vec{x}$ by moving x_i to the leftmost position. Next, note that $U_p^p \equiv \lambda x_1.(\lambda x_2.\ldots(\lambda x_p.x_p))$.

The recursive identity function $\lambda x_p.x_p$ is strictly $\lambda_{\mathcal{U}}$-definable by F_p with

$$F_p \equiv \lambda x_p.L\mathbf{I}\ulcorner 0 \urcorner x_p.$$

But then $\lambda x_{p-1}x_p.x_p$, that is the constant function $\lambda x_{p-1}.(\lambda x_p.x_p)$, can be represented by F_{p-1} with

$$F_{p-1} \equiv \lambda x_{p-1}.L(\lambda m.F_p)\ulcorner 0 \urcorner x_{p-1}.$$

We continue this process until we find that U_p^p can be represented by

$$F_1 \equiv \lambda x_0.L(\lambda m.F_1)\ulcorner 0 \urcorner x_0.$$ ◀

Together, the functions of this lemma and the previous corollary are called the initial functions.

Summarising: in this section we found that, modulo the condition that $U \in \mathcal{U}$ implies $U\mathbf{T}XY \in \mathcal{U}$ for any $X, Y \in \Lambda$, all unary total recursive functions and all initial functions are strictly $\lambda_{\mathcal{U}}$-definable. But we left open whether all p-ary total recursive functions are strictly $\lambda_{\mathcal{U}}$-definable. Hence we can not yet conclude that all partial recursive functions are strictly $\lambda_{\mathcal{U}}$-definable. To obtain this conclusion we must first show closure under primitive recursion.

3.4 Primitive recursion

▶ **Lemma 18.** *The strictly* $\lambda_{\mathcal{U}}$*-definable partial functions are closed under primitive recursion, for any set* $\mathcal{U}$ *of lambda terms satisfying the condition* $U \in \mathcal{U}$ *implies* $U\mathbf{T}XY \in \mathcal{U}$*, for any* $X, Y \in \Lambda$*.*

Proof. We can mimic the proof of [3, Lemma 6.3.7] replacing λ-definable by strictly $\lambda_{\mathcal{U}}$-definable. Since the representing term F given there is of the form **if Zero** x **then** X **else** Y, we get strictness. ◀

3.5 Undefined is satisfying certain conditions

We will show that the partial functions are *strictly* $\lambda_{\mathcal{U}}$*-definable* for any $\mathcal{U}$ satisfying certain conditions. The conditions we have seen so far are not yet enough.

▶ **Lemma 19.** *If F $\lambda_{\mathcal{U}}$-defines a p-ary partial function ϕ, then for all $n \in \mathbb{N}^p, m \in \mathbb{N}$:*

1. *$\phi(\vec{n}) = m$ iff $F\ulcorner\vec{n}\urcorner = \ulcorner m\urcorner$ and*
2. *$\phi(\vec{n})\uparrow$ iff $F\ulcorner\vec{n}\urcorner \in \mathcal{U}$,*

provided $\mathcal{U}$ satisfies the conditions:

1. *$M \notin \mathcal{U}$, for any M such that $M \twoheadrightarrow \ulcorner n\urcorner$ for some $n \in \mathbb{N}$ and*
2. *$\mathcal{U}$ is closed under reduction.*

Proof. One direction is by definition for both items.

1. If $F\ulcorner\vec{n}\urcorner = \ulcorner m\urcorner$ then, because $\ulcorner m\urcorner \notin \mathcal{U}$, $\phi(\vec{n})\downarrow$ and $\phi(\vec{n}) = m'$. But then $\ulcorner m\urcorner = \ulcorner m'\urcorner$ and hence $m = m'$. This argument is exactly as in [3, Lemma 8.4.12], but we use that $\ulcorner m\urcorner \notin \mathcal{U}$, where Barendregt uses that $\ulcorner m\urcorner$ is solvable.
2. Next suppose $F\ulcorner\vec{n}\urcorner = U \in \mathcal{U}$, then $F\ulcorner\vec{n}\urcorner \neq \ulcorner m\urcorner$ for all $m \in \mathbb{N}$. Suppose $F\ulcorner\vec{n}\urcorner = \ulcorner m\urcorner$ for some $m \in \mathbb{N}$. Then U and $\ulcorner m\urcorner$ have no common reduct, because by the conditions on $\mathcal{U}$ any reduct of U belongs to $\mathcal{U}$, while the normal form $\ulcorner m\urcorner$ does not. Therefore $\phi(\vec{n}) \neq m$ for all $m \in \mathbb{N}$. Hence $\phi(\vec{n})\uparrow$. ◀

Let us now take the standard definition of partial recursive functions as the smallest class containing the initial functions, and closed under primitive recursion and minimalisation. This is equivalent to the definition used in [3] that we repeated in Section 3.

▶ **Theorem 20.** *Let $\mathcal{U}$ be a set of lambda terms such that*

1. *$M \notin \mathcal{U}$, for any M such that $M \twoheadrightarrow \ulcorner n\urcorner$ for some $n \in \mathbb{N}$.*
2. *$\mathcal{U}$ is closed under reduction.*
3. *$\mathcal{U}$ contains all rootactive terms.*
4. *$U \in \mathcal{U}$ implies $U\mathbf{T}M_1M_2 \in \mathcal{U}$, for any $M_1, M_2 \in \Lambda$.*

Then a partial function $\phi : \mathbb{N}^p \nrightarrow \mathbb{N}$ is strictly $\lambda_{\mathcal{U}}$-definable if and only if ϕ is partial recursive.

Proof. By Corollary 16 and Lemmas 17, 11, 18 and 14, and conditions 3 and 4 on $\mathcal{U}$ it follows that all partial recursive functions are $\lambda_{\mathcal{U}}$-definable.

For the converse, assume ϕ is strictly $\lambda_{\mathcal{U}}$-definable. Then by Lemma 19 and conditions 1 and 2 on $\mathcal{U}$ we get for all $n, m \in \mathbb{N}$: $\phi(n) = m$ iff $\lambda\beta \vdash F\ulcorner n\urcorner = \ulcorner m\urcorner$. As we can recursively enumerate all conversions of the form $F\ulcorner n\urcorner = \ulcorner m\urcorner$ that can be derived in the classical lambda calculus, it follows that the graph of ϕ is recursive enumerable as well. Hence ϕ is partial recursive. ◀

4 Proposal for a definition of a set of undefined terms

In Theorem 20 we needed four conditions on $\mathcal{U}$. Let us promote them to a definition.

▶ **Definition 21.** A set $\mathcal{U}$ of lambda terms is a *set of undefined terms* if it satisfies the following conditions:

1. $M \notin \mathcal{U}$, for any M such that $M \twoheadrightarrow \ulcorner n\urcorner$ for some $n \in \mathbb{N}$.
2. $\mathcal{U}$ is closed under reduction.
3. $\mathcal{U}$ contains all rootactive terms.
4. $U \in \mathcal{U}$ implies $U\mathbf{T}M_1, M_2 \in \mathcal{U}$, for any $M_1, M_2 \in \Lambda$.

Theorem 20 implies that this collection of condition is sufficient to prove that a partial function $\phi : \mathbb{N} \nrightarrow \mathbb{N}$ is strictly $\lambda_{\mathcal{U}}$-definable if and only if ϕ is partial recursive.

We leave it open whether this set of conditions is necessary. But we consider the above set of conditions to be a reasonable first attempt at defining the concept of a set of undefined terms. Apart from condition 4, perhaps, these conditions feel quite natural.

Let us now go back to the four sets of terms $\overline{\mathcal{NF}}$, $\overline{\mathcal{HNF}}$, $\overline{\mathcal{WHNF}}$ and $\mathcal{E}$ that satisfied the condition of Statman's theorem. Note that condition 4 is implied by the condition (*) $U \in \mathcal{U}$ implies $UM \in \mathcal{U}$, for any $M \in \Lambda$. It is not difficult to see that $\overline{\mathcal{NF}}$, $\overline{\mathcal{HNF}}$ and $\overline{\mathcal{WHNF}}$ satisfy all four conditions of Definition 21. In case of the set $\mathcal{E}$ of easy terms condition 3 and (*) are well known. Hence condition 4 holds as well for $\mathcal{E}$. To show condition 1 we need a lemma, that likely belongs to folklore.

▶ **Lemma 22.** *The set $\mathcal{E}$ of easy terms is a set of undefined terms in the sense of Definition 21.*

Proof.
1. The numerals $\ulcorner n \urcorner$ are all $\beta\eta$-normal form. Hence by an application of Böhm's theorem [3, Corollary 10.4.3] they can not be easy.
2. This follows directly from the definition of easy term.
3. Condition 3 is shown in [6].
4. Condition (*) goes at least back to [12]. Hence condition 4 holds as well. ◀

5 Recap of definition of set of meaningless terms

Sets of meaningless terms were studied in the context of infinite lambda calculus. Adding infinite terms and infinite reductions that converge to a limit to the finite lambda calculus results is a calculus that is not confluent with respect to infinitary reduction. The construction of the Böhm model hints at the solution. First we add a fresh symbol $\bot$ to the syntax of finite lambda calculus and consider the set $\Lambda^{\infty}_{\bot}$ of finite and infinite λ-terms

$$M ::=_{coinduction} \bot \mid x \mid (\lambda x M) \mid (MM)$$

By Λ^{∞} we denote the subset of finite and infinite terms not containing $\bot$. Next, we choose a set $\mathcal{U} \subseteq \Lambda^{\infty}$ and add a new rule for some set $\mathcal{U} \subseteq \Lambda^{\infty}$:

$$\frac{M[\bot := \mathbf{\Omega}] \in \mathcal{U} \quad M \neq \bot}{M \to \bot} \, (\bot_{\mathcal{U}})$$

The resulting infinitary lambda calculus we denote by $\lambda^{\infty}_{\beta\bot_{\mathcal{U}}}$. We use the notation of [13]: $\twoheadrightarrow$ stands for finite reduction as in [3] and $\twoheadrightarrow\!\!\!\!\twoheadrightarrow$ stands for strongly converging (in-)finite reduction.

▶ **Definition 23** ([20]). $\mathcal{U} \subseteq \Lambda^{\infty}$ is called a *set of (finite or infinite) meaningless terms*, if it satisfies the axioms of meaninglessness:
1. *Axiom of Rootactiveness*: $\mathcal{R} \subseteq \mathcal{U}$.
2. *Axiom of Closure under β-reduction*: If $M \twoheadrightarrow\!\!\!\!\twoheadrightarrow_{\beta} N$ implies $N \in \mathcal{U}$ for all $M \in \mathcal{U}$.
3. *Axiom of Closure under Substitution*: If $M \in \mathcal{U}$ then any substitution instance of M is an element of $\mathcal{U}$.
4. *Axiom of (Weak) Overlap*: Either for each $\lambda x.P \in \mathcal{U}$, there is some $W \in \mathcal{U}$ such that $P \twoheadrightarrow\!\!\!\!\twoheadrightarrow_{\beta} Wx$, or alternatively $(\lambda x.P)Q \in \mathcal{U}$, for any $Q \in \Lambda^{\infty}_{\bot}$.
5. *Axiom of Indiscernibility*: Define $M \overset{\mathcal{U}}{\leftrightarrow} N$ if M can be transformed into N by replacing pairwise disjoint subterms of M in $\mathcal{U}$ by terms in $\mathcal{U}$. If $M \overset{\mathcal{U}}{\leftrightarrow} N$ then $M \in \mathcal{U} \Leftrightarrow N \in \mathcal{U}$.

The set Λ^∞ satisfies all these conditions. But the resulting lambda calculus is inconsistent, as all elements reduce to $\bot$. So the sets that we are interested in should be non-trivial.

▶ **Theorem 24** ([20]). *If $\mathcal{U}$ is a meaningless set, then $\lambda^\infty_{\beta\bot_\mathcal{U}}$ is confluent for infinitary β-reduction.*

For the converse we need one more condition: $\mathcal{U}$ is called closed under $\beta\bot$-expansion from $\bot$ if $N \twoheadrightarrow_{\beta\bot} \bot$ implies $N \in \mathcal{U}$ for all $N \in \Lambda^\infty$. Under this natural condition we have

▶ **Theorem 25** ([20]). *Let $\mathcal{U}$ satisfies Closure under $\beta\bot$-Expansion from $\bot$. If $\lambda^\infty_{\beta\bot_\mathcal{U}}$ is confluent, then $\mathcal{U}$ is a meaningless set.*

5.1 Sets of finite meaningless terms

There is now a mismatch: undefined terms are always finite and meaningless terms can be infinite. This can be reconciled. Instead of using the full set Λ^∞ we restrict to the closure Λ^{inf} of Λ under strongly convergent reduction. The previous two theorems hold for Λ^{inf} as well. We say that $\mathcal{U}$ is a set of *finite meaningless terms*, if its closure under strongly converging reduction is a set of meaningless terms. From now, whenever we write set of meaningless terms we mean a set of finite meaningless terms.

Let us go once more back to the four sets of terms $\overline{\mathcal{NF}}$, $\overline{\mathcal{HNF}}$, $\overline{\mathcal{WHNF}}$ and $\mathcal{E}$ that satisfied the condition of Statman's theorem. Of the four, $\overline{\mathcal{NF}}$ is the largest, and $\overline{\mathcal{HNF}}$ the second largest. Both $\overline{\mathcal{WHNF}}$ and $\mathcal{E}$[9] are subsets of $\overline{\mathcal{HNF}}$. It is well known that $\overline{\mathcal{HNF}}$ and $\overline{\mathcal{WHNF}}$ are sets of (finite) meaningless terms [14]. The other two are not:

▶ **Lemma 26.**
1. *[14] The set $\overline{\mathcal{NF}}$ does not satisfy Overlap.*
2. *The set $\mathcal{E}$ does not satisfy Indiscernibility.*

Proof.
1. $\lambda x.x\mathbf{I}\mathbf{\Omega}$ has no finite normal form, but $(\lambda x.x\mathbf{I}\mathbf{\Omega})\mathbf{K}$ reduces in two steps to the normal form $\mathbf{I}$. Note that the resulting extension infinitary term model is not consistent: $\mathbf{K} \leftarrow_\beta (\lambda x.x\mathbf{K}\mathbf{\Omega})\mathbf{K} \rightarrow_\bot \bot \leftarrow_\bot (\lambda x.x\mathbf{K}\mathbf{\Omega})\mathbf{I} \rightarrow_\beta \mathbf{I}$.
2. In [14] this was left open. But if we combine the fact that $\lambda z.\mathbf{\Omega}(\mathbf{\Theta}\lambda xyz.xzy)$ is an easy term [12][10] and the fact that $\lambda x.\Omega(xx)$ is not an easy term [7, Remark 6.2] with [21, Lemma 46(2)] (if a set of meaningless terms contains an abstraction, then it must contain all abstractions), we see that $\mathcal{E}$ can not be a set of meaningless terms. In [14] it has been shown that the first three properties hold, hence Indiscernibility does not hold. ◀

Since $\overline{\mathcal{HNF}}$ and $\overline{\mathcal{WHNF}}$ are sets of meaningless terms as well as sets of undefined terms, there is the natural question which other sets of meaningless terms can be taken as set of undefined terms. Statman's theorem is now of no help, as $\overline{\mathcal{HNF}}$ and $\overline{\mathcal{WHNF}}$ are the only sets of meaningless terms satisfying the Statman condition: the other sets of meaningless terms are not co-Visser sets. In the next section we will answer this question.

---

[9] It is straightforward to check that easy terms are unsolvable.
[10] The steps of the nice proof in [12] are: (1) $C \equiv \lambda xyz.xzy$ is right-invertible in $\lambda\beta\eta$. (2) $\mathbf{\Omega}$ is easy wrt $\lambda\beta\eta$. (3) C is easy wrt $\lambda\beta$. (4) If M is easy, then MN is easy for any N. (5) $C\mathbf{\Omega}(\mathbf{\Theta}(C\mathbf{\Omega}))$ is easy.

6 When is meaningless undefined?

▶ **Lemma 27** ([21]). *If $\mathcal{U}$ is a non-trivial set of finite or infinite meaningless terms (i.e. $\mathcal{U} \neq \Lambda^\infty$), then all its elements are unsolvable.*

▶ **Corollary 28.** *Let $\mathcal{U}$ be a non-trivial set of finite meaningless terms (i.e. $\mathcal{U} \neq \Lambda$) that satisfies Closure under $\beta\bot$-Expansion from $\bot$. Then $\mathcal{U}$ satisfies conditions 1, 2 and 3 of Definition 21.*

Proof. 2 and 3 are trivial. If $M \twoheadrightarrow \ulcorner n \urcorner$ for some $n \in \mathbb{N}$, then M reduces to a finite normal form. Hence M is solvable, because M has a head normal form. But then $M \notin \mathcal{U}$ by Lemma 27. ◀

There is a natural smallest set of meaningless terms satisfying condition 4 of Definition 21.

▶ **Definition 29.** Let us call lambda term M in Λ *almost rootactive*, if M can reduce to a term of the form $R\mathbf{T}M_1N_1 \ldots \mathbf{T}M_kN_k$ where R is rootactive and $M_i, N_i \in \Lambda$ for $1 \leq i \leq k$. Let $\mathcal{W}$ denote the set of almost rootactive terms.

Clearly the set $\mathcal{W}$ is the smallest set of *undefined* terms that satisfies the four conditions for a set of undefined terms. We also have that

▶ **Lemma 30.** *The set $\mathcal{W}$ is the smallest set of* meaningless *terms that is a set of undefined terms.*

Proof. With the techniques of [19, 21] one can show that $\mathcal{W}$ satisfies all conditions of a set of meaningless terms. ◀

7 Conclusion

We have presented a set of sufficient conditions on a set $\mathcal{U}$ of lambda terms such that a partial function $\phi : \mathbb{N}^p \nrightarrow \mathbb{N}$ is strictly $\lambda_{\mathcal{U}}$-definable if and only if ϕ is partial recursive. The smallest set $\mathcal{W}$ satisfying the these conditions is also a set of meaningless terms. By the Axiom of Indiscernibility it follows that any larger set of meaningless terms is also a set of undefined terms. Since $\mathcal{W}$ is larger than the set $\mathcal{R}$ of rootactibve terms, we conjecture that $\mathcal{R}$ can not be used to prove that a partial recursive function $\phi : \mathbb{N}^p \nrightarrow \mathbb{N}$ is strictly $\lambda_{\mathcal{R}}$-definable.

The notion of strict $\lambda_{\mathcal{U}}$-definability is forced when one searches for representations of partial recursive functions that preserve their definition. This has interesting consequences. E.g. the strict predicates suggest strongly that lambda calculus contains a many-valued logic related to McCarthy's calculus for three-valued sequential logic [8, 17].

Acknowledgements. I am grateful to Alessandro Berarducci, Alexander Kurz and the anonymous referees for helpful comments and suggestions.

References

1 H. P. Barendregt. *Some extensional term models for combinatory logics and λ-calculi.* PhD thesis, Univ. Utrecht, 1971.

2 H. P. Barendregt. A global representation of the recursive functions in the lambda-calculus. *Theor. Comput. Sci.*, 3(2):225–242, 1976. doi:10.1016/0304-3975(76)90025-6.

3 H. P. Barendregt. *The Lambda Calculus: Its Syntax and Semantics.* North-Holland, Amsterdam, Revised edition, 1984.

4 H.P. Barendregt. Solvability in lambda calculi. In M. Guillaume, editor, *Colloque international de logique: Clermont-Ferrand, 1975*, pages 209–219. Éditions du C.N.R.S., 1977.

5 H.P. Barendregt. Representing 'undefined' in lambda calculus. *J. Funct. Program.*, 2(3):367–374, 1992. doi:10.1017/S0956796800000447.

6 A. Berarducci. Infinite λ-calculus and non-sensible models. In *Logic and algebra (Pontignano, 1994)*, pages 339–377. Dekker, New York, 1996.

7 A. Berarducci and B. Intrigila. Church-Rosser lambda-theories, Infinite lambda-terms and consistency problems. In W. Hodges, M. Hyland, C. Steinhorn, and J. Truss, editors, *Logic: from Foundations to Applications*, pages 33–58. Oxford Science Publications, 1996. URL: http://www.dm.unipi.it/~berardu/Art/1996Church/CRtheories.pdf.

8 J. A. Bergstra and J. van de Pol. A calculus for four-valued sequential logic. *Theor. Comput. Sci.*, 412(28):3122–3128, 2011. doi:10.1016/j.tcs.2011.02.035.

9 A. Church. An unsolvable problem of elementary number theory. *American Journal of Mathematics*, 58(2):345–363, 1936.

10 A. Church. *The Calculi of Lambda Conversion.* Princeton University Press, 1941.

11 H. B. Curry, J. R. Hindley, and J. P. Seldin. *Combinatory Logic II.* North-Holland, 1972.

12 G. Jacopini and M. Venturini-Zilli. Easy terms in the lambda-calculus. *Fundamenta Informaticae*, VIII(2):225–233, 1985.

13 J. R. Kennaway and F. J. de Vries. Infinitary rewriting. In Terese, editor, *Term Rewriting Systems*, volume 55 of *Cambridge Tracts in Theoretical Computer Science*, pages 668–711. Cambridge University Press, 2003.

14 J. R. Kennaway, V. van Oostrom, and F. J. de Vries. Meaningless terms in rewriting. *Journal of Functional and Logic Programming*, 1999(1), 1999. URL: http://danae.uni-muenster.de/lehre/kuchen/JFLP/articles/1999/A99-01/A99-01.html.

15 S. C. Kleene. λ-definability and recursiveness. *Duke Math. J.*, 2(2):340–353, 06 1936. doi:10.1215/S0012-7094-36-00227-2.

16 S. C. Kleene. On notation for ordinal numbers. *J. Symb. Log.*, 3(4):150–155, 1938. doi:10.2307/2267778.

17 Beata Konikowska, Andrzej Tarlecki, and Andrzej Blikle. A three-valued logic for software specification and validation. *Fundam. Inform.*, 14(4):411–453, 1991.

18 G. Kreisel. Some reasons for generalising recursion theory. In R.O. Gandy and C.M.E. Yates, editors, *Logic Colloquium 1969: proceedings of the summerschool and colloquium in mathematical logic, Manchester, August 1969*, pages 139–189. North-Holland, 1971.

19 P. Severi and F. J. de Vries. Order Structures for Böhm-like models. In *Computer Science Logic*, volume 3634 of *LNCS*, pages 103–116. Springer-Verlag, 2005.

20 P. Severi and F. J. de Vries. Weakening the axiom of overlap in infinitary lambda calculus. In M. Schmidt-Schauß, editor, *Proc. of the 22nd Int'l Conf. on Rewriting Techniques and Applications, May 30 – June 1, 2011*, volume 10 of *LIPIcs*, pages 313–328. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2011. doi:10.4230/LIPIcs.RTA.2011.313.

21 P. G. Severi and F. J. de Vries. Decomposing the lattice of meaningless sets in the infinitary lambda calculus. In L. D. Beklemishev and R. de Queiroz, editors, *Proc. of the 18th Int'l Workshop on Logic, Language, Information and Computation (WoLLIC 2011)*, pages 210–227, 2011. doi:10.1007/978-3-642-20920-8_22.

22 A. M. Turing. Computability and λ-definability. *J. Symb. Log.*, 2(4):153–163, 1937. doi:10.2307/2268280.

23 A. M. Turing. The 𝔭-function in λ-K-conversion. *J. Symb. Log.*, 2(4):164, 1937. doi:10.2307/2268281.

24 A. Visser. Numerations, λ-calculus & arithmetic. In J.R. Hindley and J.P. Seldin, editors, *To H.B. Curry: Essays on Combinatory Logic, Lambda-Calculus and Formalism*, pages 259–284. Academic Press, 1980.