Graph easy sets of mute lambda terms

A. Bucciarelli[a], A. Carraro[b], G. Favro[a,b], A. Salibra[b]

[a] *Univ. Paris Diderot, Sorbonne Paris Cité, PPS, UMR 7126, CNRS, Paris, France*
[b] *DAIS, Università Ca'Foscari Venezia, Italy*

Abstract

Among the unsolvable terms of the lambda calculus, the *mute* ones are those having the highest degree of undefinedness. In this paper, we define for each natural number n, an infinite and recursive set $\mathcal{M}_n$ of mute terms, and show that it is *graph-easy*: for any closed term t of the lambda calculus there exists a graph model equating all the terms of $\mathcal{M}_n$ to t. Alongside, we provide a brief survey of the notion of undefinedness in the lambda calculus.

Keywords: Lambda-calculus, mute terms, graph models, forcing.

1. Introduction

Undefinedness in the lambda calculus is an important issue, tackled since the inception of recursion theory: in order to show that the lambda calculus is Turing-complete, a sensible notion of *undefined* λ*-term* is compulsory. In his 1936 paper showing that the Herbrand-Gödel's general recursive functions are λ-definable [31], Kleene somehow eludes this issue. First he adopts his own characterization of general recursion, obtained by closing the set of primitive recursive functions by the minimization operator μ [32]; second, when applying the μ operator to a $(k+1)$-ary primitive recursive function f, he supposes that for all k-tuples of natural numbers $n_1, \dots, n_k$ there exists a natural number m_0 such that $f(n_1, \dots, n_k, m_0) = 0$. Hence, only the *total* general recursive functions are taken into account in [31], and also in Turing's work [40], that relies on Kleene's proof of λ-definability of general recursive functions. One may wonder what happens if Kleene's λ-term implementing the μ operator is applied to a λ-term defining a unary function that never returns 0. Unsurprisingly, the resulting λ-term does not have a normal form.

Few years later, Church [21] showed that every partial recursive function f can be defined by a λ-term t_f in such a way that, whenever $f(n)$ is not defined, $t_f\ulcorner n \urcorner$ does not have a normal form, where $\ulcorner n \urcorner$ denotes the Church numeral of n. Following the terminology of Barendregt [7], Church's result may be stated as follows: every partial recursive function can be λ-defined by using the terms without normal form as undefined elements.

It is reasonable, however, that the set of λ-terms considered as undefined be a proper subset of the non-normalising ones: for instance, the Turing fixpoint combinator [41] is non-normalising, but it is very much significant from the point of view of the computations it can fire. Following Kleene and Church, several results of λ-definability of the partial recursive functions with respect to smaller and smaller sets of undefined elements have been proved. In particular, it has been shown that the following sets of λ-terms (defined in Section 2) suitably represent undefinedness: unsolvable terms [5], zero terms [38], and easy terms [42]. These results are subsumed and generalised by the following remarkable theorem of Statman [39] (see Theorem 3.5 below): every general recursive function can be λ-defined with respect to any nonempty set U of closed λ-terms, which is the complement of a recursively enumerable set (co-r.e. set, for short) and is closed under $\lambda\beta$-conversion.

Among unsolvable λ-terms, easy terms have a very flexible computational behavior. A closed λ-term t is *easy* if it can be consistently equated to any other closed term u. This notion dates back to Jacopini [25], who proved, syntactically, the easiness of $\Omega = (\lambda x.xx)(\lambda x.xx)$. Then easiness was explored by various authors, to begin with Baeten and Boerboom [4], who analysed the interpretation of Ω in graph models and proved that Ω was even graph-easy, meaning that for any other closed term u, the equation $t = u$ could even be satisfied in some graph model. In the meanwhile, the syntactic notion of a *mute* term had been introduced by Berarducci [9], correlated to an infinitary lambda calculus. Mute λ-terms aims to define a model similar to the model of Böhm trees, but which does not identify all the unsolvable terms. Mute terms are somehow the "most undefined" λ-terms, as they are zero terms, which are not $\lambda\beta$-convertible to a zero term applied to something else. For instance, Ω is mute, and $\Omega_3 = (\lambda x.xxx)(\lambda x.xxx)$ is a zero term that is not mute, since it reduces to $\Omega_3(\lambda x.xxx)$. Berarducci could argue that mute terms were the "most undefined" terms, and proved (syntactically) that the set of mute terms is an *easy-set*, meaning that all mute terms can be simultaneously and consistently equated to any given other closed term. Whether there could be a semantical proof of this result remained unexplored.

The notion of easiness can be generalised to arbitrary classes of models. Given a class $\mathcal{C}$ of models of the lambda calculus and a set Y of closed term, we say that Y is a *$\mathcal{C}$-easy set* if, for every closed λ-term u, there exists a model $\mathcal{N} \in \mathcal{C}$ in which u and all elements of Y have the same interpretation. A λ-term t is $\mathcal{C}$-easy if $\{t\}$ is a $\mathcal{C}$-easy set. Studying $\mathcal{C}$-easiness gives insights on the expressive power of $\mathcal{C}$. Concerning filter lambda models [8], for instance, it had been conjectured [3] that they have full expressive power for easy terms, in the sense that any easy term was conjectured to be filter easy. Carraro and Salibra [20] showed that this is not the case: there exists a co-r.e. set of easy terms that are not filter easy.

The first negative semantical result was obtained by Kerth [29] and shows a limitation of graph models: $\Omega_3 I$, where $I = \lambda x.x$, is an easy term, but no graph model satisfies the identity $\Omega_3 I = I$. The easiness of $\Omega_3 I$ was proven syntactically in [26] (see also [10]). A semantical proof was given only in [1], where the authors build, for each closed t, a filter model of $\Omega_3 I = t$.

Graph models are arguably the simplest models of the lambda calculus. There are two known methods for building graph models, namely: by *forcing* or by *canonical completion*. Both methods consist in completing a partial model into a total one; but, if we start with a recursive partial web, the canonical completion builds a recursive total web, while forcing always generates a non recursive web.

The canonical completion method was introduced by Plotkin and Engeler and then systematized by Longo [34] for graph models. The word "canonical" refers here to the fact that the graph model is built inductively from the partial one and completely determined by it. This method was then used by Kerth [28] to prove the existence of 2^ω pairwise inconsistent graph theories, and by Bucciarelli–Salibra [17, 15, 16] to characterize minimal and maximal graph theories. In particular [17] shows that the minimal graph theory is not equal to the minimal lambda theory $\lambda\beta$, and that the lambda theory $\mathcal{B}$ (generated by equating λ-terms with the same Böhm tree) is the greatest sensible graph theory.

The forcing method originates with Baeten–Boerboom [4] in graph models, and it is more flexible than canonical completion. In fact, the inductive construction depends here not only on the initial partial model but also on the consistency problem one is interested in. Forcing was adapted later on to a few other classes of models by Jiang [27] and Kerth [30], and generalised by Zylberajch [43] and Berline–Salibra [13] to some classes of "Omega-like terms" (cf. [11, 12] for a survey of such results).

In this paper we define an infinite and recursive set of mute terms, the *regular mute* terms. An n-ary regular mute term has the form $s_0 s_1 \dots s_n$, for some n, where each s_i is a hereditarily n-ary λ-term (Section 6). An n-ary regular mute term $s_0 s_1 \dots s_n$ has the property that, in n steps of head reduction, it reduces to a term of the same shape $s_i t_1 \dots t_n$, and the t_j are still hereditarily n-ary λ-terms. As regular mute terms are mute, we know that the set of all regular mute terms is an easy-set, since each subset of an easy set is itself easy. By generalising the forcing technique used in [13], we show that the set of n-ary regular mute terms is actually graph easy for each n. To the best of our knowledge, this is the first application of that technique to terms that do not have the shape $\Omega t_1 \dots t_n$ for some $n \geq 0$.

2. Preliminaries on lambda calculus

With regard to the lambda calculus we follow the notation and terminology of [6]. By Λ and Λ^o, respectively, we indicate the set of λ-terms and of closed λ-terms. The symbol $\equiv$ denotes syntactical equality. The following are some notable λ-terms that will be used throughout the paper:

- $I \equiv \lambda x.x; \qquad K \equiv \lambda xy.x; \qquad \omega \equiv \lambda x.xx;$
- $\Omega \equiv \omega\omega; \qquad \omega_3 \equiv \lambda x.xxx; \qquad \Omega_3 \equiv \omega_3\omega_3;$
- $\mathbf{Y} \equiv \lambda f.(\lambda x.f(xx))(\lambda x.f(xx)),$ the Curry fixed point combinator.

A λ-term has exactly one of the following forms:

(i) $\lambda x_1 \dots x_n.yt_1 \dots t_k$ $(n, k \geq 0)$;

(ii) $\lambda x_1 \dots x_n.(\lambda y.s)ut_1 \dots t_k$ $(n, k \geq 0)$.

The first term is a *head normal form* (hnf, for short). The redex $(\lambda y.s)u$ in the second one is called a *head redex.*

A λ-term t is *solvable* if it has a hnf u, i.e., t is β-convertible to a λ-term u and u is a hnf. A term is *unsolvable* if it is not solvable.

An unsolvable t has: (i) order 0 if it is not β-convertible to an abstraction; (ii) order ∞ if, for every natural number $n > 0$, t is β-convertible to $\lambda x_1 \dots x_n.u$ for some u; (iii) order $n \geq 1$ if there exists a greatest positive number n such that t is β-convertible to $\lambda x_1 \dots x_n.u$ for some u. Terms of order 0 are also called *zero terms.* For example, Ω has order 0, $\lambda x.\Omega$ has order 1 and $\mathbf{Y}K$ has order ∞.

A λ-*theory* is a congruence on Λ (with respect to the operators of abstraction and application) which contains $\alpha\beta$-conversion. We denote by $\lambda\beta$ the least λ-theory. A λ-theory is *consistent* if it does not equate all λ-terms, *inconsistent* otherwise. The set of λ-theories constitutes a complete lattice with respect to inclusion, whose top is the inconsistent λ-theory and whose bottom element is the theory $\lambda\beta$.

If T is a λ-theory and t is a λ-term, we denote by $[t]_T$ the equivalence class of t.

Definition 2.1. *Given a λ-theory T, a set A of λ-terms is T*-closed *if*

$$t \in A \wedge t =_T u \Rightarrow u \in A.$$

A set A of λ-terms is T-closed if and only if $A = \bigcup_{t \in A} [t]_T$.

Definition 2.2. *If X is a set of equations between λ-terms, then the* λ-theory $\mathrm{Th}(X)$ generated by X *is defined as the intersection of all λ-theories containing X.*

Notation 2.3. If $Y \subseteq \Lambda^o$, then we define $\mathrm{Th}(Y)$ to be the λ-theory $\mathrm{Th}(\{(t,u) : t,u \in Y\})$.

Definition 2.4. *We say that $Y \subseteq \Lambda^o$ is* consistent *if $\mathrm{Th}(Y)$ is consistent.*

The definition of what is in general a "model of λ-calculus" can be found in many books and papers (see, for example, [6, 33, 11]). We recall here that every model $\mathcal{N}$ of the λ-calculus induces a λ-theory, denoted by $\mathrm{Th}(\mathcal{N})$, called the *equational theory* of $\mathcal{N}$. Thus, $t = u \in \mathrm{Th}(\mathcal{N})$ if and only if t and u have the same interpretation in $\mathcal{N}$.

If $Y \subseteq \Lambda^o$, then we say that $\mathcal{N}$ *satisfies* Y, and we write $\mathcal{N} \models Y$, if $\mathrm{Th}(Y) \subseteq \mathrm{Th}(\mathcal{N})$.

3. Undefinedness in lambda calculus

The problem of characterizing λ-terms that represent an undefined computational process has interested researchers since the origin of lambda calculus. In this section we survey some results about this issue.

3.1. Unsolvable terms

If t is a solvable λ-term, then the principal hnf of t is obtained by applying to t the head reduction: $t \twoheadrightarrow_h \lambda x_1 \dots x_n.yu_1 \dots u_k$. The context $\lambda x_1 \dots x_n.y[\ ]_1 \dots [\ ]_k$ can be considered as a first partial result of the computation of t. The Böhm tree of t is the possibly infinite unfolding of t according to the following coinductive definition:

$$\mathrm{BT}(t) = \begin{cases} \bot & \text{if } t \text{ is unsolvable;} \\ \lambda x_1 \dots x_n.y\mathrm{BT}(u_1) \dots \mathrm{BT}(u_k) & \text{if } t \twoheadrightarrow_h \lambda x_1 \dots x_n.yu_1 \dots u_k. \end{cases}$$

The Böhm tree represents the idea of "asymptotic behavior", where the possible computational content of unsolvables is not taken into account, for they have all the empty Böhm tree.

If the operational semantics of a λ-term is its Böhm tree it is natural that the terms representing undefinedness are the unsolvable ones. The Genericity Lemma of lambda calculus (see [6, Proposition 14.3.24]) provides evidence for the lack of operational meaning of unsolvables: for every unsolvable t and normal form u, from $C[t] =_{\lambda\beta} u$ it follows that $C[s] =_{\lambda\beta} u$ for every $s \in \Lambda^o$.

All unsolvables can be identified and still we get a consistent λ-theory. Theories in which all unsolvables are identified are called *sensible* and have been extensively studied in literature (see [6, Chapter 16]). There is a continuum of sensible theories and it can be naturally equipped with a structure of bounded lattice. The λ-theory $\mathcal{H}$, generated by equating all unsolvables, is the minimal sensible theory and it admits one maximal extension $\mathcal{H}^*$. The most renowned sensible theory is the theory $\mathcal{B} = \{t = u : \mathrm{BT}(t) = \mathrm{BT}(u)\}$ of Böhm trees.

3.2. A fine classification of undefinedness

We may classify the order of undefinedness of a set of closed λ-terms according to the "size" of the set of terms it can be consistently equated to.

Let Y be a set of closed λ-terms. Define

$$\mathrm{Con}(Y) = \{u \in \Lambda^o : \ Y \cup \{u\} \text{ is consistent (see Definition 2.4)}\}.$$

If $Y = \{t\}$ is a singleton set, we write $\mathrm{Con}(t)$ for $\mathrm{Con}(\{t\})$.

It is obvious that $Y \subseteq \Lambda^o$ is consistent if and only if $\mathrm{Con}(Y) \neq \emptyset$. The case in which $\mathrm{Con}(Y) = \Lambda^o$ is perhaps the most studied in the λ-calculus.

Definition 3.1. *A set $Y \subseteq \Lambda^o$ is an* easy-set *if* $\mathrm{Con}(Y) = \Lambda^o$.

A term t is *easy* if $\{t\}$ is an easy-set.

Easy terms were introduced in [25], where it is also shown using syntactical techniques that Ω is easy. Semantical proofs of easiness originated in [4].

The set of easy terms is a proper co-r.e. subset of the unsolvables. For example, Ω_3 is unsolvable but not easy, because it cannot be consistently equated to the identity I. Although Ω_3 is not easy, it is possible to show that $\Omega_3 I$ is easy (see [26]).

Definition 3.2. *A term t is* nf-easy *if* $\mathrm{Con}(t) \supseteq \{u \in \Lambda^o : u \textit{ is a normal form}\}$.

In [10] Berarducci and Intrigila prove many interesting results on easy terms. Some of these results are collected in the following theorem that shows the unusual behavior of easy terms.

We recall that the Böhm tree of a term t is incompatible with that of a term u if there exists no λ-term s such that $BT(t) \subseteq BT(s)$ and $BT(u) \subseteq BT(s)$.

Theorem 3.3. *[10] The following assertions hold:*

1. *There exists $t \in \Lambda^o$ such that* $\mathrm{Con}(t) = \Lambda^o \setminus [I]_{\lambda\beta}$.
2. *There exists a nf-easy term that is not easy.*
3. *A term t is easy iff* $\mathrm{Con}(t) \supseteq \{u \in \Lambda^o : \mathrm{BT}(u) \textit{ is finite}\}$.
4. $u \in \mathrm{Con}(\mathbf{Y}\Omega_3)$ *for every closed term u such that $BT(u) \not\subseteq BT(\omega_3)$.*

It is not yet known whether $\mathbf{Y}\Omega_3$ is easy ([14]).

Any element of an easy-set is obviously an easy term. Berline-Salibra [13] have shown that the set $\{\Omega(\lambda x_0 \dots x_{k+1}.x_{k+1}) : k \in \mathbb{N}\}$ is an easy-set. There exist sets of easy terms that are not easy-sets: easiness of $\{\Omega, \Omega I\}$ fails because $\{\Omega, \Omega I, K\}$ is not consistent. In particular, the set of all easy terms is not an easy-set ([24]).

3.3. Statman-sets

In this section we introduce Statman-sets, which are the most suitable candidates for representing inside lambda calculus the undefined value of a partial recursive function.

For every natural number n, we denote by $\ulcorner n \urcorner$ the Church numeral of n.

Definition 3.4. *A $\lambda\beta$-closed set $B \subseteq \Lambda^o$ is a* Statman-set *if, for every partial recursive function $f : \mathbb{N} \to \mathbb{N}$, there exists $F \in \Lambda^o$ such that*

$$\begin{cases} F\ulcorner n \urcorner =_{\lambda\beta} \ulcorner f(n) \urcorner & \text{if } f \downarrow n; \\ F\ulcorner n \urcorner \in B & \text{otherwise.} \end{cases}$$

Statman has shown the following beautiful result in an unpublished paper [39]. The proof by Statman is based on early results by Visser [42] and can be found in [7].

Theorem 3.5. *[7, Theorem 4.1] Every nonempty co-r.e. $\lambda\beta$-closed set of closed λ-terms is a Statman-set.*

Since $\mathrm{Con}(A)$ is a co-r.e. set for every $\lambda\beta$-closed r.e. set A, we get the following proposition.

Proposition 3.6. *Let $A \subsetneq \Lambda^o$ be a $\lambda\beta$-closed r.e. set. Then we have:*

1. $\Lambda^o \setminus A$ *is a Statman-set.*
2. $\mathrm{Con}(A)$ *is a Statman-set for every* $\mathrm{Con}(A) \neq \emptyset$.
3. $\mathrm{Con}(t)$ *is a Statman-set for every closed λ-term t.*

Example 3.7. *The set of closed λ-terms without normal form is a Statman-set. The same holds for the set of unsolvable (resp. easy, zero) closed terms.*

3.4. Mute terms and meaningless sets

Berarducci trees [9] take into account the computational content of the unsolvables. As for Böhm trees, Berarducci trees are obtained by an infinite unfolding of λ-terms.

A *top normal form* (top-nf, for short) is either a variable or an abstraction or a zero term applied to another term. A term t has a top-nf if t is $\lambda\beta$-convertible to a top-nf.

The Berarducci tree of a term t is the possibly infinite unfolding of t according to the following coinductive definition:

$$\mathrm{BD}(t) = \begin{cases} \bot & \text{if } t \text{ has no top-nf;} \\ x & \text{if } t =_{\lambda\beta} x; \\ \lambda x.\mathrm{BD}(u) & \text{if } t =_{\lambda\beta} \lambda x.u; \\ \mathrm{BD}(s) \cdot \mathrm{BD}(u) & \text{if } t =_{\lambda\beta} su \text{ with } s \text{ zero term.} \end{cases}$$

The function BD is well-defined by [9, Theorem 9.5].

As an example, we build the Berarducci tree of Ω_3. The only possible reduction path of Ω_3 is the following:

$$\Omega_3 \to_\beta \Omega_3\omega_3 \to_\beta \Omega_3\omega_3\omega_3 \to_\beta \Omega_3\omega_3\omega_3\omega_3 \to_\beta \dots,$$

where $\omega_3 \equiv \lambda x.xxx$. So the zero term Ω_3 can be seen as the infinite term $((\dots\omega_3)\omega_3)\omega_3$.

One of the main results of [9] is that the λ-theory $\mathrm{BD} = \{t = u : \mathrm{BD}(t) = \mathrm{BD}(u)\}$ of Berarducci trees is consistent. The terms that have a bottom Berarducci tree are called *mute* terms and can be formally defined as follows.

Definition 3.8. *[9] A term t is mute if t has no top-nf.*

Mute terms have a totally undefined operational behavior. As expressed in the next theorem, mute terms are "so undefined" that they satisfy the strongest property of undefinedness we have introduced so far.

Theorem 3.9. *[9] The set of mute terms is an easy-set.*

All trees defined in this section can be also defined as infinite normal forms of an infinitary extension of β-reduction. The infinite normal form $NF^{\infty}(t)$ of a term t is the Böhm tree of t if the infinitary reduction is made up to identifications of all unsolvables, or the Berarducci tree of t if the infinitary reduction is made up to identifications of all mute terms.

If we do not identify a suitable set of unsolvables the infinitary calculus is in general not confluent as the following example shows. The mute term tt, where $t = \lambda x.I(xx)$, reduces to $I(I(I(\ldots)))$ with an infinite reduction and to Ω with a finite one; but then it is not possible to close the diagram by reducing $I(I(I(\ldots)))$ to Ω.

The lack of confluence in infinitary lambda calculus originated the analysis of the so-called *meaningless sets*. Informally, a set A of unsolvables is "meaningless" if the confluence of infinitary lambda calculus can be restored by identifying all its elements.

The formal definition of a meaningless set can be found in [36, 37]. We have a lattice of meaningless sets whose top element is the set of all unsolvables and whose bottom element is the set of all mute terms. Every meaningless set A of unsolvables defines a notion of A-tree that determines a consistent λ-theory. It has been shown in [36] that there exists a continuum of different theories determined by meaningless sets.

3.5. Semantics

The following definition introduces the semantical notion of easiness.

Definition 3.10. *Let $Y \subseteq \Lambda^o$ and $\mathcal{C}$ be a class of models of the λ-calculus. We say that Y is a* $\mathcal{C}$-easy set *if, for every closed λ-term t, there exists a model $\mathcal{N} \in \mathcal{C}$ such that $\mathcal{N} \models Y \cup \{t\}$. A λ-term t is $\mathcal{C}$-easy if $\{t\}$ is a $\mathcal{C}$-easy set.*

The notion of $\mathcal{C}$-easiness is useful to analyse the expressive power of a class of models.

In this paper we will focus on graph easiness, namely on $\mathcal{C}$-easiness, where $\mathcal{C}$ is the class of graph models (see Section 4). The origin of this technique dates back to the proof of graph easiness of Ω in [4]. The main concern of this paper will be the definition of a family $\mathcal{M}_k$ ($k \in \mathbb{N}$) of infinite sets of mute terms, the regular ones, and a semantical proof that, for each k, the set $\mathcal{M}_k$ is a graph easy-set.

The reader can consult Section 8 for a list of positive and negative results concerning the $\mathcal{C}$-easiness of some classes of models.

4. Graph models

The class of graph models belongs to Scott's continuous semantics (see [11, 12]). Graph models owe their name to the fact that continuous functions are encoded in them via (a sufficient fragment of) their graphs, namely their traces.

A graph model is a model of the untyped lambda calculus, which is generated from a web in a way that will be recalled below. Historically, the first graph model was Plotkin and Scott's P_ω (see e.g. [6]), which is also known in the literature as "the graph model". The simplest graph model was introduced soon afterwards, and independently, by Engeler [22] and Plotkin [35]. More examples can be found in [11].

As a matter of notation, we denote by D^* the set of all finite subsets of a set D. Elements of D^* will be denoted by small roman letters $a, b, c, \ldots$, while elements of D by Greek letters $\alpha, \beta, \gamma, \ldots$.

For short we will confuse the model and its web and so we define:

Definition 4.1. *A* graph model *is a pair (D, p), where D is an infinite set and $p : D^* \times D \to D$ is an injective total function.*

Such a pair will also be called a *total pair*. In the setting of graph models a *partial pair* (see [11]) is a pair (A, q) where A is any set and $q : A^* \times A \rightharpoonup A$ is a partial (possibly total) injection. Examples of partial pairs are: the empty pair $(\emptyset, \emptyset)$ and all the graph models.

If (D, p) is a partial pair, we write $a \to_p \alpha$ (or $a \to \alpha$ if p is evident from the context) for $p(a, \alpha)$. Moreover, $\beta \to \alpha$ means $\{\beta\} \to \alpha$. The notation $a_1 \to a_2 \to \cdots \to a_{n-1} \to a_n \to \alpha$ stands for $(a_1 \to (a_2 \to \ldots (a_{n-1} \to (a_n \to \alpha)) \ldots))$. If $\bar{a} = a_1, a_2, \ldots, a_n$, then $\bar{a} \to \alpha$ stands for $a_1 \to a_2 \to \ldots a_{n-1} \to a_n \to \alpha$.

A total pair (D, p) generates a model of the λ-calculus of universe $\mathcal{P}(D)$, called *graph λ-model*.

We denote by $\mathrm{Env}_{\mathcal{P}(D)}$ the set of environments, i.e., the functions from the set V of λ-calculus variables to $\mathcal{P}(D)$. For every environment ρ, $x \in \mathrm{V}$ and $a \subseteq D$, we denote by $\rho[x := a]$ the new environment ρ' which coincides with ρ everywhere except on x, where ρ' takes the value a.

The interpretation $|t|^p : \mathrm{Env}_{\mathcal{P}(D)} \to \mathcal{P}(D)$ of a λ-term t relative to (D, p) can be described inductively as follows:

- $|x|^p_\rho = \rho(x)$
- $|tu|^p_\rho = \{\alpha : (\exists a \subseteq_{\mathrm{fin}} |u|^p_\rho) \;\; a \to \alpha \in |t|^p_\rho\}$
- $|\lambda x.t|^p_\rho = \{\, a \to \alpha : \alpha \in |t|^p_{\rho[x:=a]}\}$

Since $|t|^p_\rho$ only depends on the value of ρ on the free variables of t, we write $|t|^p$ if t is closed.

A graph model (D, p) *satisfies* $t = u$, written $(D, p) \vDash t = u$, if $|t|^p_\rho = |u|^p_\rho$ for all environments ρ. The λ-theory $\mathrm{Th}(D, p)$ induced by (D, p) is defined as

$$\mathrm{Th}(D, p) = \{t = u : t, u \in \Lambda \text{ and } |t|^p_\rho = |u|^p_\rho \text{ for every } \rho\}.$$

A λ-theory induced by a graph model will be called a *graph theory*.

It is well known that the semantics of lambda calculus given in terms of graph models is incomplete, since it trivially omits the axiom of extensionality. In [17] Bucciarelli and Salibra show the existence of a graph model (D, p), whose theory $\mathrm{Th}(D, p) \neq \lambda\beta$ is the least graph theory with respect to the inclusion among theories.

5. Forcing in graph model

An important tool that we use in Section 7 is a technique introduced by Baeten and Boerboom in [4], known as "forcing in graph models". In [4] they gave the first semantical proof of the easiness of Ω: for any term t, they build a graph model (D, p) such that $(D, p) \models t = \Omega$. Their graph model is defined by a method of forcing, which, although much simpler than the forcing techniques used in set theory, is somewhat in the same spirit.

The essence of the forcing technique is the following: membership of some element of the web to the interpretation of some adequate λ-term (here Ω) is *forced* by some condition on p. The following proposition clarifies what we have just said. It gives a necessary condition and a sufficient condition for an element to be in the interpretation of Ω in a graph model.

Proposition 5.1. *([4]) Let (D, p) be a graph model and $\alpha \in D$.*

1. *If $\alpha \in |\Omega|^p$, then there exists a finite subset a of D such that $a \to_p \alpha \in a$;*
2. *If there exists β such that $\beta = \beta \to_p \alpha$, then $\alpha \in |\Omega|^p$.*

The conditions expressed in Proposition 5.1 are the most important technical tools used in [4] to solve the problem of the easiness of Ω from the semantical side.

Once the conditions on p, matching a given purpose, have been found, the following challenge is to build a graph model satisfying those conditions; this is achieved by starting from a suitable partial pair, and carefully completing it to a total pair. The completion must be careful in order to preserve the desired property in the total pair. This is achieved with an *ad hoc* construction in [4]. A generalisation of this construction, involving a notion of weakly continuous function and presented below as Theorem 5.4, has been proposed in [13]. In Section 7 we will generalise it further, proving Theorem 7.4, which is the key of the graph-easiness of the set of n-regular mute terms. Let us explain now what "weakly continuous" means in this context.

Notation 5.2. Let D be an infinite countable set. By $\mathcal{I}(D)$ we indicate the cpo of partial injections $q : D^* \times D \rightharpoonup D$, ordered by inclusion of their graphs.

By a "*total q*" we will mean "an element of $\mathcal{I}(D)$ which is a total map" (equivalently: which is a maximal element of $\mathcal{I}(D)$). The domain and range of $q \in \mathcal{I}(D)$ are denoted by $dom(q)$ and $rg(q)$. We will also confuse the partial injections and their graphs.

Definition 5.3. *[13, Definition 10] A function $F : \mathcal{I}(D) \to \mathcal{P}(D)$ is* weakly continuous *if it is monotone with respect to inclusion and if furthermore, for all total $p \in \mathcal{I}(D)$,*

$$F(p) = \bigcup_{q \subseteq_{\text{fin}} p} F(q).$$

Since we are working with a countable infinite D, the difference with continuity comes of course from the fact that there exist elements of $\mathcal{I}(D)$ which are not total but of infinite cardinality.

The forcing completion process we were referring to is the core of the proof of Theorem 5.4 below, which is the fundamental tool to prove the graph easiness of Ω.

Theorem 5.4. *[13, Theorem 11] If $F : \mathcal{I}(D) \to \mathcal{P}(D)$ is weakly continuous, then there exists a total p such that $|\Omega|^p = F(p)$.*

On the other hand, one has:

Lemma 5.5. *[13, Lemma 15] For every closed λ-term t, the function $F_t : \mathcal{I}(D) \to \mathcal{P}(D)$, defined by $F_t(q) = \{\alpha \in D \ : \ \forall \text{ total } p \supseteq q,\ \alpha \in |t|^p\}$, is weakly continuous, and we have $F_t(p) = |t|^p$ for each total p.*

The fact that for any closed term t there exists a total $p \in \mathcal{I}(D)$ such that $|t|^p = |\Omega|^p$, that is the *graph easiness* of Ω, is a simple corollary of the two results above.

6. The regular mute λ-terms

The aim of this section is to define the set of n-regular mute λ-terms (see Definition 6.4). A first step will be to introduce the set of hereditarily n-ary λ-terms (see Definition 6.1). An n-regular mute term has the form $s_0 s_1 \dots s_n$, where s_i is a hereditarily n-ary λ-term. An n-regular mute term has the property that, in n steps of head reduction, it reduces to a term of the same shape $t_0 t_1 \dots t_n$, where $t_0 = s_i$ for some $1 \leq i \leq n$. We show that regular mute terms are mute, so that the set of all regular mute terms is an easy-set, as subset of an easy-set. We will show that, for each n, the set of n-regular mute terms is actually a graph-easy set, by generalising the forcing technique used in [13].

A first step towards the definition of regular mute terms is the definition of the hereditarily n-ary terms.

Definition 6.1. *Let V be the infinite set of variables of λ-calculus and $n \geq 1$. The set $H_n[V]$ of* hereditarily n-ary terms *(over V) is the smallest set of λ-terms containing V and such that: For all $t_1, \dots, t_n \in H_n[V]$, distinct variables $y_1, \dots, y_n \in V$ and $i \leq n$ we have: $\lambda y_1 \dots y_n . y_i t_1 \dots t_n \in H_n[V]$.*

We denote by $H_n[\bar{x}]$, for $\bar{x}$ any finite (and possibly empty) sequence of distinct variables in V, the set of terms of $H_n[V]$ whose free variables are included in $\bar{x}$. We write H_n for $H_n[]$.

Notice that

(i) The hereditarily n-ary terms are normal forms.

(ii) $t \in H_n[\bar{x}]$ iff either t is a variable in $\bar{x}$ or there exists a sequence $\bar{y}$ of fresh distinct variables such that $t = \lambda y_1 \dots y_n . y_i t_1 \dots t_n$, where $t_j \in H_n[\bar{x}, \bar{y}]$.

Example 6.2. *Some hereditarily unary and binary λ-terms:*

(i) $\lambda y.yy \in H_1$

(ii) $\lambda y.yx \in H_1[x]$; $(x \in H_1[x, y])$

(iii) $\lambda x.x(\lambda y.yx) \in H_1$; $(\lambda y.yx \in H_1[x])$

(iv) $\lambda zy.yzx \in H_2[x]$; $(z, x \in H_2[x, y, z])$

(v) $\lambda xy.x(\lambda zt.tzx)y \in H_2$; $(\lambda zt.tzx), y \in H_2[x, y])$.

If $\bar{x}$ is a sequence, then $l(\bar{x})$ denotes the length of the sequence $\bar{x}$.

Lemma 6.3. (Closure of $H_n[V]$ under substitution) *Let $n \in \omega$ and $t \in H_n[V]$. Then:*

(i) *For all $\bar{z} \in V$ and all $\bar{u} \in H_n[V]$ such that $l(\bar{u}) = l(\bar{z})$, we have $t[\bar{u}/\bar{z}] \in H_n[V]$.*

(ii) *Moreover if $t \in H_n[\bar{x}, \bar{z}]$ and $\bar{u} \in H_n[\bar{x}]$, then $t[\bar{u}/\bar{z}] \in H_n[\bar{x}]$.*

Proof. One gets (i) via a trivial induction on t; then (ii) trivially follows.

Definition 6.4. *An* n-regular mute term *is a term* $s_0 s_1 \dots s_n$ *such that* $s_i \in H_n$*. The set of all* n*-regular mute terms is denoted by* $\mathcal{M}_n$*.*

Example 6.5. *Some unary and binary regular mute terms:*

- $(\lambda x.xx)(\lambda x.xx) \in \mathcal{M}_1$
- $(\lambda x.x(\lambda y.yx))(\lambda x.xx) \in \mathcal{M}_1$
- $AAA \in \mathcal{M}_2$*, where* $A := \lambda xy.x(\lambda zt.tzx)y$*.*

Example 6.6. *Let* $B := \lambda x.x(\lambda y.xy)$*. Then* BB *is a mute term that is not regular:*

$$BB = (\lambda x.x(\lambda y.xy))B \to_\beta B(\lambda y.By) \to_\beta BB$$

In the following we denote by $\to_h$ the head reduction (see [6, Definition 8.3.10]). We write $\to_h^n$ for n steps of head reduction.

Proposition 6.7. *For every* $s_0 s_1 \dots s_n \in \mathcal{M}_n$ *there exist* $r_1, \dots, r_n \in H_n$ *and* $1 \leq i \leq n$ *such that*

$$s_0 s_1 \dots s_n \to_h^n s_i r_1 \dots r_n.$$

Proof. Since $s_0 \in H_n$, then $s_0 \equiv \lambda y_1 \dots y_n . y_i t_1 \dots t_n$ with $t_1, \dots, t_n \in H_n[y_1, \dots, y_n]$. Hence $s_0 s_1 \dots s_n \to_h^n s_i t_1[\bar{s}/\bar{y}] \dots t_n[\bar{s}/\bar{y}]$, where $\bar{y} \equiv y_1 \dots y_n$ and $\bar{s} \equiv s_1 \dots s_n$. By Lemma 6.3 the term $t_i[\bar{s}/\bar{y}] \in H_n$, and we are done by defining $r_i \equiv t_i[\bar{s}/\bar{y}]$.

Theorem 6.8. *Every* n*-regular mute term is mute.*

Proof. Let $s_0 \dots s_n$ be a n-regular mute term. By Proposition 6.7 there exists an infinite sequence i_k $(k \geq 1)$ of natural numbers and an infinite head reduction path

$$s_0 \dots s_n \to_h^n s_{i_1} r_1 \dots r_n \to_h^n r_{i_2} t_1 \dots t_n \to_h^n \dots$$

that has an infinite number of redex at the top of the term. Since $s_0 \dots s_n$ cannot reach a top normal form, then it is a mute term.

Theorem 6.9. $\mathcal{M}_n$ *is a recursive set of mute terms.*

Proof. The recursiveness of $\mathcal{M}_n$ trivially follows from that of H_n, which is clear.

7. Forcing for regular mute terms

In this section we show that, given a closed λ-term t and a natural number n, there exists a graph model which equates all the n-regular mute terms to t, using forcing.

7.1. Some useful lemmas

Lemma 7.2 below generalises Proposition 5.1 obtained by Baeten and Boerboom in [4] and gives a sufficient condition for an element to be in the interpretation of an n-regular mute term in a graph model.

Lemma 7.1. *Let (D, p) be a graph model, ρ be an environment, $\beta \in D$, and $\bar{\beta} = \beta, \beta, \dots, \beta$ (n-times). If $\beta = \bar{\beta} \to \alpha$, $t \in H_n[\bar{x}]$ and $\beta \in \rho(x_i)$ $(i = 1, \dots, k)$ then $\beta \in |t|^p_\rho$.*

Proof. The proof is by induction over the complexity of t as hereditarily n-ary λ-term. If $t \equiv x_i$ then the conclusion is trivial because $\beta \in \rho(x_i)$. Otherwise, there exists $\bar{u} \equiv u_1, \dots, u_n \in H_n[\bar{x}, \bar{y}]$ such that $t = \lambda\bar{y}.y_i\bar{u}$.

$$\beta = \bar{\beta} \to \alpha \in |\lambda\bar{y}.y_i\bar{u}|^p_\rho \ \Leftrightarrow \ \alpha \in |y_i\bar{u}|^p_{\rho[\bar{y}:=\bar{\beta}]}.$$

Since $\beta \in \rho[\bar{y} := \bar{\beta}](y_i)$ for every $i = 1, \dots, n$, then by induction hypothesis $\beta \in |u_i|^p_{\rho[\bar{y}:=\bar{\beta}]}$ for every $i = 1, \dots, n$. It follows that $\alpha \in |y_i\bar{u}|^p_{\rho[\bar{y}:=\bar{\beta}]}$ and we get the conclusion.

Lemma 7.2. *Let (D, p) be a graph model, $t \in \mathcal{M}_n$ and $\gamma \in |t|^p$. Then there exist a sequence $\beta_i \equiv a^i_1 \to \dots \to a^i_n \to \gamma$ $(i \in \omega)$ of elements of D and a sequence d_i $(i \in \omega)$ of natural numbers $\leq n$ such that $\beta_{i+1} \in a^i_{d_i}$.*

Proof. Let $t = s^0_0 s^0_1 \dots s^0_n$, where $s^0_i \in H_n$. By Proposition 6.7 there exists an infinite sequence of mute terms such that

$$s^0_0 s^0_1 \dots s^0_n \ \to^n_\beta \ s^1_0 s^1_1 \dots s^1_n \ \to^n_\beta \ \dots \ \to^n_\beta \ s^k_0 s^k_1 \dots s^k_n \ \to^n_\beta \dots$$

and $s^k_0 \equiv s^{k-1}_{d_{k-1}}$ for some $1 \leq d_{k-1} \leq n$. The number d_{k-1} is the order of the head variable of the term s^{k-1}_0. By $\gamma \in |s^0_0 s^0_1 \dots s^0_n|^p$ there exists $a^0_1 \to \dots \to a^0_n \to \gamma \in |s^0_0|^p$ such that $a^0_i \subseteq |s^0_i|^p$. We define

$$\beta_0 = a^0_1 \to \dots \to a^0_n \to \gamma.$$

Assume $\beta_k = a^k_1 \to \dots \to a^k_n \to \gamma \in |s^k_0|^p$ and $a^k_j \subseteq |s^k_j|^p$ for every $1 \leq j \leq n$. Since $\beta_k \in |s^k_0|^p$ and $s^k_0 = \lambda\bar{y}.y_{d_k} u_1 \dots u_n$ for some terms $u_i \in H_n[\bar{y}]$, then we have

$$\gamma \in a^k_{d_k} |u_1[\bar{a}/\bar{y}]|^p \dots |u_n[\bar{a}/\bar{y}]|^p,$$

where $\bar{a} = a^k_1, \dots, a^k_n$. It follows that there exists

$$\beta_{k+1} = a^{k+1}_1 \to \dots \to a^{k+1}_n \to \gamma \in a^k_{d_k} \tag{1}$$

such that $a^{k+1}_j \subseteq |u_j[\bar{a}/\bar{y}]|^p$. We have to prove that $\beta_{k+1} \in |s^{k+1}_0|^p$ and $a^{k+1}_j \subseteq |s^{k+1}_j|^p$ for every $j \leq n$. By applying the induction hypothesis and (1) we get $\beta_{k+1} \in a^k_{d_k} \subseteq |s^k_{d_k}|^p = |s^{k+1}_0|^p$. The other relation can be obtained as follows, by defining $\bar{s}^k = s^k_1, \dots, s^k_n$: $a^{k+1}_j \subseteq |u_j[\bar{a}/\bar{y}]|^p \subseteq |u_j[|\bar{s}^k|^p/\bar{y}]|^p = |s^{k+1}_j|^p$.

7.2. Forcing at work

Let $\mathcal{I}(D)$ be the cpo of partial injection from $D^* \times D$ into D. If $p \in \mathcal{I}(D)$ then the *universe* $\mathrm{Univ}(p)$ of p is defined as follows:

$$\mathrm{Univ}(p) = \bigcup_{(a,\alpha)\in dom(p)} (a \cup \{\alpha, p(a,\alpha)\}).$$

If p is finite, then the universe of p is also finite.

Definition 7.3. *Let $p \in \mathcal{I}(D)$ be finite, $\alpha \in D$ and $\bar{\epsilon} \equiv \epsilon_1, \dots, \epsilon_k \in D \setminus \mathrm{Univ}(p)$. Then $p_{\bar{\epsilon},\alpha}$ is the extension of p such that $\epsilon_j = \epsilon_1 \to \epsilon_{j+1}$ $(j = 1, \dots, k)$, where $\epsilon_{k+1} = \alpha$.*

Notice that

$$\epsilon_1 = \epsilon_1 \to \epsilon_1 \to \cdots \to \epsilon_1 \to \alpha \quad (k\text{-times})$$

and $p_{\bar{\epsilon},\alpha}$ is also finite.

The next theorem is the main technical tool for proving the graph-easiness of the set of n-regular mute terms. It generalises [13, Thm. 11].

Theorem 7.4. *Let $F : \mathcal{I}(D) \to \mathcal{P}(D)$ be a weakly continuous function and let $e \in \mathbb{N}$. Then there exists a total $p' : D^* \times D \to D$ such that $(D, p') \models t = F(p')$ for all e-regular mute terms t.*

Proof. The proof of this theorem will be concluded just before Theorem 7.9.

We are going to build an increasing sequence of finite injective maps $p_n : D^* \times D \rightharpoonup D$, starting from $p_0 = \emptyset$, and a sequence of elements $\alpha_n \in D \cup \{*\}$, where $*$ is a new element, such that: $p' =_{\mathrm{def}} \cup p_n$ is a total injection, and $(D, p') \models t = F(p') = \{\alpha_n : n \in \omega\} \cap D$, for all $t \in \mathcal{M}_e$.

We fix an enumeration of D and an enumeration of $D^* \times D$.

We start from $p_0 = \emptyset$.

Assume that $p_n : D^* \times D \rightharpoonup D$ and $\alpha_0, \dots, \alpha_{n-1}$ have been built. We let

- $\alpha_n =$ First element of $F(p_n) \setminus \{\alpha_0, \dots, \alpha_{n-1}\}$ in the enumeration of D, if this set is non-empty, and $\alpha_n = *$ otherwise;
- $(b_n, \delta_n) =$ First element in $(D^* \times D) \setminus dom(p_n)$;
- $\gamma_n =$ First element in $D \setminus (\mathrm{Univ}(p_n) \cup b_n \cup \{\delta_n\} \cup \{\alpha_0, \dots, \alpha_{n-1}, \alpha_n\})$.

We define a new finite injection r as follows:

$$r(\beta) = \begin{cases} p_n(\beta) & \text{if } \beta \in dom(p_n) \\ \gamma_n & \text{if } \beta = (b_n, \delta_n) \end{cases}$$

Case 1: $\alpha_n = *$. We let $p_{n+1} = r$.

Case 2: $\alpha_n \in D$. We define $p_{n+1} = r_{\bar{\epsilon}^n,\alpha_n}$ (see Definition 7.3), where $\bar{\epsilon}^n = \epsilon_1^n, \dots, \epsilon_e^n \in D \setminus (\mathrm{Univ}(r) \cup \{\alpha_n\})$ are the first e distinct elements of $D \setminus (\mathrm{Univ}(r) \cup \{\alpha_n\})$.

It is clear that p_n is a strictly increasing sequence of well-defined finite injective maps and that $p' = \cup p_n$ is total.

It is also clear that each p_n (and p') is partitioned into two disjoint sets: $p_n = p_n^1 \cup p_n^2$, where $p_n^1 = \{b_i \to \delta_i = \gamma_i : 1 \le i \le n-1\}$ is called the *gamma part* of p_n and $p_n^2 = p_n \setminus p_n^1$ is called the *epsilon part*.

For every $\gamma \in D$, we define

$$deg(\gamma) = \begin{cases} 0 & \text{if } \gamma \notin rg(p') \\ min\{n : \gamma \in rg(p_n)\} & \text{if } \gamma \in rg(p') \end{cases}$$

Moreover, $deg(c) = max\{deg(x) : x \in c\}$ for every $c \subseteq_{\text{fin}} D$.

The following claims easily derive from the construction of p'.

Claim 7.5. $\forall n' > n,\ (rg(p_{n'}) \setminus rg(p_n)) \cap \mathrm{Univ}(p_n) = \emptyset$.

Proof. Let $\beta \in rg(p_{n'}) \setminus rg(p_n)$ and $k = deg(\beta)$. Then $n < k \le n'$. Since $\beta \in rg(p_k) \setminus rg(p_{k-1})$ and β can be either γ_{k-1} or one of the ϵ_j^{k-1}, then β is not an element of $\mathrm{Univ}(p_{k-1})$ by construction of p_k. Since $\mathrm{Univ}(p_n) \subseteq \mathrm{Univ}(p_{k-1})$, we get the conclusion.

Claim 7.6. *If $deg(a \to \alpha) = n$ and $\alpha \notin rg(p_n)$, then $\alpha \notin rg(p')$.*

Proof. If $\alpha \in rg(p')$ then $\alpha \in rg(p_j)$ for some j. By $\alpha \notin rg(p_n)$, it must be $n < j$. We also have that $p'(a, \alpha) = p_n(a, \alpha)$, for $deg(p'(a, \alpha)) = n$. Thus $(a, \alpha) \in dom(p_n)$ and $\alpha \in \mathrm{Univ}(p_n)$. By Claim 7.5 and $\alpha \in rg(p_j) \setminus rg(p_n)$, we get a contradiction.

Claim 7.7. *(i) $deg(a \to \alpha) \ge deg(a \cup \{\alpha\})$.*

(ii) If $a \to \alpha$ is in the gamma part of p', then $deg(a \to \alpha) > deg(a \cup \{\alpha\})$.

Proof. Let $deg(a \to \alpha) = n$. Thus, $p'(a, \alpha) = p_n(a, \alpha)$ and $a \cup \{\alpha\} \subseteq \mathrm{Univ}(p_n)$.

(i) If $n < deg(\alpha) = j$, then $\alpha \in rg(p_j) \setminus rg(p_n)$, that contradicts Claim 7.6. It follows that $deg(\alpha) \le n$.

If $n < deg(a) = j$, then there exists $\theta \in a$ such that $deg(\theta) = j > n$, so that $\theta \in rg(p_j) \setminus rg(p_n)$. By Claim 7.5 $\theta \notin \mathrm{Univ}(p_n)$. This contradicts $a \subseteq \mathrm{Univ}(p_n)$.

(ii) By (i) it is sufficient to show that $deg(a \cup \{\alpha\}) \ne n$. By hypothesis $a = b_{n-1}$, $\alpha = \delta_{n-1}$ and $p_n(a, \alpha) = \gamma_{n-1}$. Then by construction $deg(\gamma_{n-1}) = n$. By definition of p_n, α is different from γ_{n-1} and from any ϵ_j^{n-1}. So it cannot be in $rg(p_n) \setminus rg(p_{n-1})$. The same reasoning applies to $a = b_{n-1}$.

Claim 7.8. *If $\alpha_n \in rg(p')$, then $deg(\alpha_n) \le n$.*

Proof. By construction of p' we have that $\alpha_n \in \mathrm{Univ}(p_{n+1})$ and $\alpha_n \notin rg(p_{n+1}) \setminus rg(p_n)$. Then $deg(\alpha_n) \ne n+1$. If $deg(\alpha_n) = j > n+1$, then $\alpha_n \in rg(p_j) \setminus rg(p_{n+1})$, which contradicts Claim 7.5 because $\alpha_n \in \mathrm{Univ}(p_{n+1})$. This concludes the proof of Claim 7.8.

We now show that $(D, p') \models t = \{\alpha_n : n \in \omega\} \cap D = F(p')$, for every $t \in \mathcal{M}_e$.

Let $X = \{\alpha_n : n \in \omega\} \cap D$ and $t \equiv s_0 s_1 \dots s_e \in \mathcal{M}_e$ in the remaining part of this proof.

- $X \subseteq F(p')$: It follows from the definition of α_n and from the fact that $F(p_n) \subseteq F(p')$.

- $F(p') \subseteq X$: Suppose $\gamma \in F(p')$; since F is weakly continuous, $\gamma \in F(p_i)$ for some i (and for all the larger ones). If $\gamma \notin X$ then, for all $n \geq i$, $\alpha_n \neq *$ has rank smaller than γ in the enumeration of D, contradicting the fact that there is only a finite number of such elements.

- $X \subseteq |t|^{p'}$: Let $\alpha_n \neq *$. The condition $(D, p') \models \alpha_n \in |t|^{p'}$ follows immediately from Lemma 7.1 and the fact that

$$\epsilon_1^n = \epsilon_1^n \to \epsilon_1^n \to \cdots \to \epsilon_1^n \to \alpha_n \quad (e\text{-times}).$$

- $|t|^{p'} \subseteq X$: Assume that $\gamma \in |t|^{p'}$. Then by Lemma 7.2 there exists a sequence $\beta_j \equiv a_1^j \to \cdots \to a_e^j \to \gamma$ $(j \in \omega)$ of elements of D and a sequence d_j $(j \in \omega)$ of natural numbers $\leq e$ satisfying the property $\beta_{j+1} \in a_{d_j}^j$. By Claim 7.7 and by $\beta_{j+1} \in a_{d_j}^j$ the sequence $deg(\beta_j)$ is an infinite decreasing sequence of natural numbers. Then there exist i and n such that $deg(\beta_k) = n + 1$ for all $k \geq i$.

  There are (at most) $e + 1$ elements having degree $n + 1$, namely

  $\gamma_n = b_n \to \delta_n$

  $\epsilon_1^n = \epsilon_1^n \to \cdots \to \epsilon_1^n \to \alpha_n$ (e-times)

  $\epsilon_2^n = \epsilon_1^n \to \cdots \to \epsilon_1^n \to \alpha_n$ ($e-1$-times)

  ...

  $\epsilon_e^n = \epsilon_1^n \to \alpha_n$.

  Since $deg(\beta_i) = n + 1$, then β_i is one of the element listed above, too. We have the following possibilities:

  (1): $\beta_i \equiv \gamma_n = b_n \to \delta_n$ is not possible. In fact, by the definition of β_i we derive that $b_n \equiv a_1^i$ and $\delta_n \equiv a_2^i \to \cdots \to a_e^i \to \gamma$. By Claim 7.7(ii) $deg(b_n)$ and $deg(\delta_n)$ are strictly less than $n+1$, so that $deg(a_j^i) < n+1$ for every $1 \leq j \leq e$. From $\beta_{i+1} \in a_{d_i}^i$ we get the contradiction $deg(\beta_{i+1}) < n + 1$.

  Then we must have that $\beta_i \equiv \epsilon_r^n$ for some r.

  (2) $\beta_i \equiv \epsilon_r^n = \epsilon_1^n \to \cdots \to \epsilon_1^n \to \alpha_n$ ($e - r + 1$-times). By the definition of β_i we derive that $a_j^i = \{\epsilon_1^n\}$ for every $1 \leq j \leq e - r + 1$ and $\alpha_n \equiv a_{e-r+2}^i \to \cdots \to a_e^i \to \gamma$. But by Claim 7.8 we have that $deg(\alpha_n) < n + 1$. This implies that $\beta_{i+1} = \epsilon_1^n$ and then that $\gamma = \alpha_n$.

This concludes the proof of Theorem 7.4.

Theorem 7.9. *Let t be a closed term. Then, for every natural number n there exists a graph model (D, p') such that $(D, p') \models t = u$ for all n-regular mute terms $u \in \mathcal{M}_n$.*

Proof. Take (D, p') as defined in Theorem 7.4, using F_t as defined in Lemma 5.5.

8. Conclusion and Future Work

We have defined a family $\mathcal{M}_k$ ($k \in \mathbb{N}$) of infinite, recursive sets of mute terms, the regular ones, and shown that for each k, the set $\mathcal{M}_k$ is a graph-easy set: for any closed λ-term t there exists a graph model which equates t and all the k-ary regular mutes.

In a broader perspective, we have provided a positive answer to an instance of the following problem:

Given a set E of closed λ-terms and a class $\mathcal{C}$ of models of λ-calculus, is it the case that E is a $\mathcal{C}$-easy set?

We refer to this problem as the one of the $\mathcal{C}$-easiness of E. If $E = \{t\}$ is a singleton set we refer to the $\mathcal{C}$-easiness of t. Few instances of this problem have been solved so far:

1. graph-easiness of Ω, proved in [4]. Generalisations of the techniques of [4] led to several new proofs of graph-easiness in [43] and [13].
2. graph-easiness of the set $\{\Omega(\lambda x_1 \ldots x_k.x_k) : k \geq 1\}$, proved in [13].
3. filter-easiness of $\Omega_3 I$ and some other easy terms, proved in [1], using the framework set up in [3].
4. $\mathcal{C}$-easiness of Ω for the class $\mathcal{C}$ of the reflexive coherent spaces (in stable semantics), proved in [30].

A very restricted number of instances of the problem of $\mathcal{C}$-easiness has obtained a negative answer so far:

5. the graph-easiness of $\Omega_3 I$ disproved in [29], where it has been shown that no graph model satisfies the identity $\Omega_3 I = I$.
6. It has been conjectured in [3] that every easy term is filter-easy. This problem became popular as the nineteenth item of the *TLCA list of open problems* and has been recently disproved in [20], where it was shown the existence of a nonempty co-r.e. set of easy terms which are not filter-easy.

The landscape obtained considering these results in a unified framework, suggests several research directions. Some examples:

- It is unknown whether $Y\Omega_3$ is easy. Is there a suitable class of models $\mathcal{C}$ such that $Y\Omega_3$ is $\mathcal{C}$-easy? Otherwise stated: is it possible to attack the problem of the easiness of $Y\Omega_3$ from the semantical side? Generalising techniques *à la* forcing in graph models to other classes of models, like for instance the relational ones [18], could be promising.

- Is there a non-syntactical class $\mathcal{C}$ of models, such that t is $\mathcal{C}$-easy, for every easy term t?

- More directly related to the content of this work is the following:

  Is $\bigcup_n \mathcal{M}_n$ a graph-easy set?

  Using a technique developed in [19] we are able to prove that the answer is *yes*, if $\bigcup_{n \in E} \mathcal{M}_n$ is a graph-easy set for each finite set E.

  Nevertheless, for $n_1 < n_2$, dealing with $\mathcal{M}_{n_1}$ and $\mathcal{M}_{n_2}$ simultaneously is problematic in our approach, since the elements $\epsilon_1, \ldots, \epsilon_{n_2}$, as defined in the proof of Theorem 7.4, force new, unwanted elements to belong to the interpretation of the n_1-regular mutes. A simpler question, embodying the difficulty of the problem, concerns the graph-easiness of $\mathcal{M}_1 \cup \mathcal{M}_2$.

[1] Alessi, F., Dezani-Ciancaglini, M., Honsell, F.: Filter models and easy terms, Italian Conference on Theoretical Computer Science, LNCS 2202, Springer-Verlag, 17–37, 2001.

[2] Alessi, F. , Dezani-Ciancaglini, M., Lusin, S. : Intersection types and domain operators, Theoretical Computer Science, 316(1), 25–47, 2004.

[3] Alessi, F., Lusin, S.: Simple easy terms, in S. van Bakel (ed.), Intersection Types and Related Systems, ENTCS 70, Elsevier, 1–18, 2002.

[4] Baeten, J., Boerboom, B.: Omega can be anything it should not be, in Proceedings of the Koninklijke Nederlandse Akademie van Wetenschappen, Serie A, Indag. Mathematicae 41, 111–120, 1979.

[5] Barendregt, H.P.: Some extensional term models for lambda calculi and combinatory logics, PhD Thesis, Utrecht University, 1971.

[6] Barendregt, H.P.: The lambda-calculus, its syntax and semantics, Studies in Logic vol. 103, North_Holland, revised edition 1984.

[7] Barendregt, H.P.: Representing 'undefined' in lambda calculus. J. Functional Programming 2(3): 367–374, 1992.

[8] Barendregt, H., Coppo, M., Dezani-Ciancaglini, M.: A filter model and the completeness of type assignment, Journal of Symbolic Logic 48(4), p.931-940, 1983.

[9] Berarducci, A.: Infinite λ-calculus and non-sensible models, in Logic and Algebra, eds. A. Ursini and P. Agliano, Lecture Notes in Pure and Applied Mathematics 180, Marcel Dekker Inc., 339–377, 1996.

[10] Berarducci, A., Intrigila, B.: Some new results on easy λ-terms, Theoretical Computer Science 121, 71–88, 1993.

[11] Berline, C.: From computation to foundations via functions and application: The lambda-calculus and its webbed models, Theoretical Computer Science 249, 81–161, 2000.

[12] Berline, C.: Graph models of λ-calculus at work, and variations. Mathematical Structures in Computer Science 16, 185–221, 2006.

[13] Berline, C., Salibra, A.: Easiness in graph models, Theoretical Computer Science 354, 4–23, 2006.

[14] Bertini Y.: La notion d'indéfini en lambda-calcul. PhD thesis, Université de Savoie, 2005.

[15] Bucciarelli, A., Salibra, A.: The minimal graph-model of lambda-calculus, 28th Internat. Symp. on Math. Foundations of Comput. Science, LNCS 2747, Springer-Verlag, 300–307, 2003.

[16] Bucciarelli, A., Salibra, A.: The sensible graph theories of lambda-calculus, Proceedings of the 19th Annual Symposium on Logic in Computer Science (LICS'04), IEEE, 276–285, 2004.

[17] Bucciarelli, A., Salibra, A.: Graph lambda-theories, Mathematical Structures in Computer Science 18(5), 975–1004, 2008.

[18] Bucciarelli, A., Ehrhard, T., Manzonetto, G.: Not Enough Points Is Enough. in CSL 2007, LNCS 4646, 298-312, Springer 2007.

[19] Bucciarelli, A., Carraro, A., Salibra, A.: Minimal lambda-theories by ultraproducts. In D. Kesner and P. Viana: Proceedings Seventh Workshop on Logical and Semantic Frameworks, with Applications (LSFA 2012), EPTCS 113, 61–76, 2012.

[20] Carraro, A., Salibra, A.: Easy lambda-terms are not always simple, RAIRO - Theor. Inform. and Applic. 46(2), 291–314, 2012.

[21] Church, A.: The calculi of lambda conversion. Princeton University Press, 1941.

[22] Engeler, E.: Algebras and combinators, Alg. Univ. 13(3), 289–371, 1981.

[23] Hasegawa, M.: Recursion from cyclic sharing: Traced monoidal categories and models of cyclic lambda calculi. In P. de Groote (ed.), Proceedings of Typed Lambda Calculi and Applications (TLCA'97), LNCS 1210, 196–213, 1997.

[24] Intrigila B.: A problem on easy terms in lambda calculus. Fund. Inform, 15(1), 99–106, 1991.

[25] Jacopini, C.: A condition for identifying two elements of whatever model of combinatory logic, in C. Böhm, ed., LNCS 37, Springer Verlag, 213–219, 1975.

[26] Jacopini, C., Venturini-Zilli, M.: Easy terms in the lambda-calculus, Fundamenta Informaticae VIII.2, 225–233, 1985.

[27] Jiang, Y.: Consistency of a λ-theory with n-tuples and easy terms, Archives of Math. Logic, 34(2), 79–96, 1995.

[28] Kerth, R.: Isomorphism and equational equivalence of continuous λ-models, Studia Logica 61, 403–415, 1998.

[29] Kerth, R.: Isomorphisme et équivalence équationnelle entre modèles du λ-calcul, Thèse, Université Paris 7, 1995.

[30] Kerth, R.: Forcing in stable models of untyped λ-calculus, Indagationas Mathematicae 10 , 59–71, 1999.

[31] Kleene, S.C.: λ-definability and recursiveness, Duke Math. J. 2, 340–353, 1936.

[32] Kleene, S.C.: General recursive functions of natural numbers, Math. Ann. 112, 727–742, 1936.

[33] Krivine, J. L.: Lambda-Calculus, Types and Models. Masson, Paris, and Ellis Horwood, Hemel Hempstead, 1993.

[34] Longo, G.: Set-theoretical models of λ-calculus : theories, expansions and isomorphisms, Annals of Pure and Applied Logic 24, 153–188, 1983.

[35] Plotkin, G.: A set-theoretical definition of application, Memorandum MIP-R-95, School of artificial intelligence, University of Edinburgh, 1972.

[36] Severi, P.G., de Vries, F.J.: Weakening the Axiom of Overlap in the Infinitary Lambda Calculus. In RTA, LIPICS vol. 10, 313–328, 2011.

[37] Severi, P.G., de Vries, F.J.: Decomposition and cardinality of intervals in the lattice of meaningless sets, in WOLLIC, LNAI 6642, Springer, 210–227, 2011.

[38] Statman, R.: Two recursion theoretic problems in lambda calculus. Manuscript, Department of Mathematics, Carnegie Mellon University, Pittsburgh, USA, 1987.

[39] Statman, R.: The Visser Fixed Point Theorem. Manuscript, Department of Mathematics, Carnegie Mellon University, Pittsburgh, USA, 1990.

[40] Turing, A.: Computability and λ-definability. J. Symbolic Logic 2(4), 153–163, 1937.

[41] Turing, A.: The þ-function in λ-K-conversion. J. Symbolic Logic 2(4), 164, 1937.

[42] Visser, A.: Numerations, λ-calculus and arithmetic, To H.B. Curry: Essays on Combinatory Logic, Lambda-Calculus and Formalism (J.R. Hindley and J.P. Seldin eds.), Academic Press, New York, p. 259-284, 1980.

[43] Zylberajch, C.: Syntaxe et sémantique de la facilité en λ-calcul, Thèse, Université Paris 7, 1991.