Available online at www.sciencedirect.com

ScienceDirect

Electronic Notes in Theoretical Computer Science 303 (2014) 59–77

Electronic Notes in Theoretical Computer Science

www.elsevier.com/locate/entcs

Towards a Notion of Lambda Monoid

Martin Hyland[1,2]

Department of Pure Mathematics and Mathematical Statistics
University of Cambridge
Cambridge, U.K.

Abstract

Any interpretation of the lambda calculus determines a composition monoid and this monoid can be equipped with structure from which the interpretation can be recovered. That is the essence of Dana Scott's account of the lambda calculus in terms of its category of retracts. This paper presents a new approach to the needed structure on the monoid deriving from a recent analysis of the lambda calculus in terms of algebraic theory.

Keywords: Algebraic theories, abstract clones, λ-calculus, reflexive objects, Λ-monoids.

1 Introduction

We owe the insight that interpretations of the lambda calculus can be understood in terms of reflexive objects in cartesian closed categories to the fertile mind of Dana Scott. The idea became well established in circles close to him before eventually being outlined in his paper [10] for the Curry Festschrift. Scott's approach was to consider the category of retracts arising from a composition monoid associated with an interpretation of the λ-calculus. This prompted others (Karst Koymans in [7] and Jim Lambek and Phil Scott in [8]) to analyse what was the structure on the composition monoid which ensures that the category of retracts is cartesian closed. The answers which they arrived at were probably very close though that is not obvious from what can be found in the literature. In this paper I outline a different approach.

Lambek and Scott [8] concentrate on a notion of C-monoid which is very strong: it corresponds to having a (perhaps one should say) non-trivial object U of a carte-

[1] I thank John Power for organising the Workshop on Algebra, Coalgebra and Topology at Bath as part of the Wessex Theory Seminar and for encouragement to write a paper. I thank Mike Behrsich, Sebastian Kerkhoff and Martin Schneider of TU Dresden for advice and discussion about abstract clones at and after the meeting.

[2] Email: M.Hyland@dpmms.cam.ac.uk

http://dx.doi.org/10.1016/j.entcs.2014.02.004
1571-0661/© 2014 Elsevier B.V. All rights reserved.

sian closed category with specified isomorphisms $U \times U \cong U \cong U^U$. In showing the existence of such a situation Lambek and Scott exploit an elegant categorical reduction of the problem due to Dana Scott. (In passing I remark that the account of the intellectual history in [8] is misleading. The Dana Scott's clever reduction was not as Lambek and Scott (and others following their lead) say the original construction. A direct domain theory construction, which is sketched in 2.5.11 of Paul Taylor's Thesis [11], was found first in 1973. I still think that it has independent interest.) Lambek and Scott's focus on the very special situation of a C-monoid was natural in terms of their book, but they were aware of weaker possibilities: exercises in [8] give a notion of weak C-monoid. This is close to what Koymans in his detailed analysis [7] calls a cartesian closed monoid. Koymans realised that the connection between the monoid with structure and the interpretation of the λ-calculus was not at all straightforward and much of [7] is concerned to tie that down. In tightening the connection he arrived at his final notion which he called lambda monoid or λ-monoid. H

Very roughly the idea of the approaches both of Lambek and Scott and of Koymans was to code up what will be products and then give a suitably related operation of lambda abstraction. There are a collection of equations whose meaning is perhaps not completely evident. In this paper I present a fresh approach which avoids explicit treatment of lambda abstraction, but an analogue of the coding of products remains. The choice of a coding for products raises issues even for the case of C-monoids: the matter is treated in the exercises of [8]. The issue was taken much further by Koymans in [7]. It seems that the significance his analysis is not widely appreciated. In this paper I present a new take on the structure on the composition monad. I hope by means of it to make the considerations addressed by Koymans more transparent.

To tie down his notion of λ-monoid, Koymans had to write down a lot of equations. In presenting an alternative approach I am not trying to reduce the number of equations: I shall certainly have more. Rather I am seeking a formulation whose meaning is evident in general terms and which specifically makes sense of what I call the Fundamental Theorem of the Lambda Calculus. I have explained this theorem in a recent paper [5]. There I presented a reading of interpretations of the λ-calculus as algebraic theories with structure. Breaking with established usage I call these λ-theories. The more standard idea of an interpretation, the old notion of λ-algebra as in [2], is rather technical. It corresponds in my setting to the more superficially more accessible notion of an algebra for the initial λ-theory Λ, that is, to a Λ-algebra.

The key argument in [5] passes from a Λ-algebra to a corresponding monoid and then to the presheaf category on that monoid. The presheaf category is equipped with a reflexive object and to establish the Fundamental Theorem one needs some detailed information. In [5] all the needed calculations are done directly in the lambda calculus: I had no space to develop a more abstract treatment of the monoid. In this paper I explain from the point of view of [5] what is the essential structure on the monoid. I do not try to present a definitive notion of Λ-monoid. I would

rather do that at the same time as making good a serious omission from [5], that is, the connection with combinatory logic. It will not perhaps be immediately obvious but in fact the structure of Λ-monoid is intimately related to the algebraic understanding of the combinators, and the issues detected by Koymans are reflected there. As I hope to show on another occasion all this can be understood from the perspective of algebraic theories and an approach to the structure on the monoid which arises naturally from the perspective of abstract clones [12]. In this paper I restrict myself to explaining that perspective, and in terms of it to give some hints about the analysis in [7].

2 Algebraic theories

By the account given in [5] an interpretation of the λ-calculus is a λ-theory, that is, by definition an algebraic theory with additional structure. For this paper it seems worth putting that in context by reviewing three notions of algebraic theory. I start with a definition which incorporates the general perspective, explained in [6], that various kinds of algebraic theory are given by monads in some Kleisli bicategory [3]. I shall then unpick the definition a little to arrive at a more familiar presentation and finally I shall make clear the connection with the idea of an abstract clone [12]. This material is essentially folklore in the customary sense that it is fairly evident albeit not clearly discussed in the literature. The reason for giving an account here is this. The notion of λ-theory is expressed using the categorical notion of algebraic theory. On the other hand the notion of Λ-monoid which I want to outline is best understood as derived from a fragment of an algebraic theory, with the latter best thought of as an abstract clone.

2.1 Cartesian operads

I give first a definition of algebraic theory exactly parallel to that for operads or symmetric operads. As with the definition of operads there is a lot of structure to lay out. Of course that is an argument in favour of the abstract approach of [6].

Write **Sets** for the category of sets and $\mathbb{F}$ for a standard skeleton of the category of finite sets, whose objects are, if you will, the finite cardinals. The operadic approach focuses on the functor

$$\mathbb{F}^2 \to \mathbb{F}; \qquad (n, m) \mapsto n + m$$

which is in this special case the coproduct.

The operadic approach is notation heavy and it is as well to set some of it up in advance. We shall need some notation for multiple coproducts. For a sequence

$$\mathbf{m} = (m_1, \cdots, m_n)$$

of objects of $\mathbb{F}$, let

$$\Sigma\mathbf{m} = m_1 + \cdots + m_n$$

be the sum as an object of $\mathbb{F}$. Similarly for a sequence

$$\mathbf{g} = (g_1 : m_1 \to p_1, \cdots, g_n : m_n \to p_n)$$

of maps in $\mathbb{F}$, let

$$\Sigma\mathbf{g} : \Sigma\mathbf{m} \to \Sigma\mathbf{p}$$

be the evident map arising from the functoriality of $+$.

We also need some notation to handle reindexing. Suppose that we have

$$\mathbf{p} = (p_1, \cdots p_m)$$

an m-ary sequence of objects of $\mathbb{F}$ and a map $f : n \to m$ in $\mathbb{F}$. Then we have the induced n-ary sequence

$$f^*\mathbf{p} = (p_{f(1)}, \cdots p_{f(n)}) \,.$$

The fact that $+$ is a coproduct gives an obvious induced map

$$\hat{f} : \Sigma f^*\mathbf{p} \to \Sigma\mathbf{p} \,.$$

Now for the definition of an algebraic theory. I am afraid that further notation gets introduced along the way.

Definition 2.1 (Operadic version) A *cartesian operad* $\mathcal{T}$ is given by the following data:

- a functor $\mathcal{T} : \mathbb{F} \to \mathbf{Sets}$;
- a choice of identity element $\mathbf{id} \in \mathcal{T}(1)$;
- for each n and $m_1, \cdots m_n$ a composition map

$$\mathcal{T}(n) \times (\mathcal{T}(m_1) \times \cdots \times \mathcal{T}(m_n)) \longrightarrow \mathcal{T}(m_1 + \cdots + m_n) \,,$$

  written $(f, g_1, \cdots, g_n) \mapsto f[g_1, \cdots, g_n]$.

This data is required to satisfy certain unit and associative laws together with naturality conditions.

- The *unit laws* are the simple equations

$$\mathbf{id}[t] = t \,; \qquad t[\mathbf{id}, \cdots, \mathbf{id}] \,.$$

- To give the *associative law* one needs some further notation. For $\mathbf{m} = (m_1, \cdots m_n)$ a sequence of objects of $\mathbb{F}$ let

$$\mathcal{T}(\mathbf{m}) = \mathcal{T}(m_1) \times \cdots \times \mathcal{T}(m_n) \,.$$

  Note that composition can now be written as

$$\mathcal{T}(n) \times \mathcal{T}(\mathbf{m}) \longrightarrow \mathcal{T}(\Sigma\mathbf{m}) \,.$$

Suppose that we have an object n, a sequence of objects $\mathbf{m} = (m_1, \cdots, m_n)$ and for each $1 \leq i \leq n$ a sequence $\mathbf{p_i} = (p_{i1}, \cdots, p_{im_i})$. There is an evident parallel composition

$$\mathcal{T}(\mathbf{m}) \times \prod_i \mathcal{T}(\mathbf{p_i}) \longrightarrow \prod_i (\mathcal{T}(m_i) \times \mathcal{T}(\mathbf{p_i})) \longrightarrow \prod_i \mathcal{T}(\Sigma \mathbf{p_i}) \,.$$

This involves some evident reordering which I do not spell out. The $\Sigma \mathbf{p_i}$ themselves form a sequence with sum $\Sigma_i(\Sigma \mathbf{p_i})$, say. Then the associative law is the requirement that the diagram

$$\begin{tikzcd}
\mathcal{T}(n) \times \mathcal{T}(\mathbf{m}) \times \prod_i \mathcal{T}(\mathbf{p_i}) \arrow[r] \arrow[d] & \mathcal{T}(\Sigma \mathbf{m}) \times \prod_i \mathcal{T}(\mathbf{p_i}) \arrow[d] \\
\mathcal{T}(n) \times \prod_i \mathcal{T}(\Sigma \mathbf{p_i}) \arrow[r] & \mathcal{T}(\Sigma_i \Sigma \mathbf{p_i})
\end{tikzcd}$$

commutes. Here the parallel composition introduced above appears on the left. The effect of associativity is that the notation

$$t[s_1[r_{11}, \cdots, r_{1m_1}], \cdots, s_n[r_{n1}, \cdots, r_{nm_n}]]$$

is unambiguous.

- The first and straightforward naturality condition is this. Suppose that we have an object n and maps $g_1 : m_1 \to p_1$, ... , $g_n : m_n \to p_n$. Then the diagram

$$\begin{tikzcd}
\mathcal{T}(n) \times \mathcal{T}(\mathbf{m}) \arrow[r] \arrow[d, "\mathcal{T}(n) \times \mathcal{T}(\mathbf{g})"'] & \mathcal{T}(\Sigma \mathbf{m}) \arrow[d, "\mathcal{T}(\Sigma \mathbf{g})"] \\
\mathcal{T}(n) \times \mathcal{T}(\mathbf{p}) \arrow[r] & \mathcal{T}(\Sigma \mathbf{p})
\end{tikzcd}$$

commutes. The notation $\Sigma \mathbf{g}$ was explained above and the meaning of $\mathcal{T}(\mathbf{g})$ is parallel to that of $\mathcal{T}(\mathbf{m})$ above.

- The second naturality condition is bit more fiddly. Suppose that we have a map $f : n \to m$ and a sequence $\mathbf{p} = p_1, \cdots p_m$ of objects. The basic properties of products give an evident map $\mathcal{T}(\mathbf{p}) \to \mathcal{T}(f^* \mathbf{p})$. The naturality condition involves also the notation $\hat{f}$, introduced before the definition. It is that the diagram

$$\begin{tikzcd}
\mathcal{T}(n) \times \mathcal{T}(\mathbf{p}) \arrow[r] \arrow[d, "\mathcal{T}(f) \times \mathcal{T}(\mathbf{p})"'] & \mathcal{T}(n) \times \mathcal{T}(f^* \mathbf{p}) \arrow[r] & \mathcal{T}(\Sigma f^* \mathbf{p}) \arrow[d, "\mathcal{T} \hat{f}"] \\
\mathcal{T}(m) \times \mathcal{T}(\mathbf{p}) \arrow[rr] & & \mathcal{T}(\Sigma \mathbf{p})
\end{tikzcd}$$

commutes.

That concludes the operadic definition. It is in the spirit of Linear Logic. The basic operations are linear in the sense that there is no weakening or duplication but these structural rules are handled within the functoriality $\mathbb{F} \to \mathbf{Sets}$.

2.2 Algebraic theories

Suppose that we have a cartesian operad $\mathcal{T}$. We can derive some further structure as follows. First, for each of the n points $k : 1 \to n$ of an object $n \in \mathbb{F}$, we have an element $\mathbf{pr}_k = k_*\mathbf{id} \in \mathcal{T}(n)$, the image of the identity $\mathbf{id} \in \mathcal{T}(1)$. Secondly for each n and m we can define a simultaneous composition with identification of variables $\mathcal{T}(n) \times \mathcal{T}(m)^n \to \mathcal{T}(m)$ corresponding to the traditional notion of substitution: we have an evident projection map $\mathbf{snd} : n \times m \to m$ and we can take the composite

$$\mathcal{T}(n) \times \mathcal{T}(m)^n \longrightarrow \mathcal{T}(n \times m) \xrightarrow{\mathbf{snd}} \mathcal{T}(m)\,.$$

This leads to a more familiar formulation which I hope will be regarded as canonical.

Definition 2.2 (Category theoretic version) An *algebraic theory* $\mathcal{T}$ is given by the following data:

- a functor $\mathcal{T} : \mathbb{F} \to \mathbf{Sets}$;
- for each $n \in \mathbb{F}$, projections $\mathbf{pr}_1, \cdots, \mathbf{pr}_n \in \mathcal{T}(n)$;
- composition maps $\mathcal{T}(n) \times \mathcal{T}(m)^n \to \mathcal{T}(m); (f, g_1, \cdots, g_n) \mapsto f(g_1, \cdots, g_n)$.

This data is required to satisfy the following laws.

- *Unit laws* The identity $\mathbf{id} = \mathbf{pr}_1 \in \mathcal{T}(1)$ is the special projection in $\mathcal{T}(1)$ and the other projections are defined by $\mathbf{pr}_k = k_*\mathbf{id} \in \mathcal{T}(n)$. The identity acts as the identity in that $\mathbf{id}(f) = f$; moreover there is a compatibility with projections for composition on the other side $f(\mathbf{pr}_1, \cdots, \mathbf{pr}_n) = f$.
- *Associative law* For all n, m and p the diagram

$$\begin{tikzcd}
\mathcal{T}(n) \times \mathcal{T}(m)^n \times \mathcal{T}(p)^m \arrow[rr] \arrow[d] & & \mathcal{T}(m) \times \mathcal{T}(p)^m \arrow[d] \\
\mathcal{T}(n) \times (\mathcal{T}(m) \times \mathcal{T}(p)^m)^n \arrow[r] & \mathcal{T}(n) \times \mathcal{T}(p)^n \arrow[r] & \mathcal{T}(p)\,,
\end{tikzcd}$$

  commutes, where the left hand arrow is the evident duplication of arguments.
- *Naturality* The composition $\mathcal{T}(n) \times \mathcal{T}(m)^n \to \mathcal{T}(m)$ is natural in m and in n in

the sense that for $h : m \to p$ and for $f : n \to k$ the diagrams

$$\begin{tikzcd} \mathcal{T}(n) \times \mathcal{T}(m)^n \arrow[r] \arrow[d] & \mathcal{T}(m) \arrow[d] \\ \mathcal{T}(n) \times \mathcal{T}(p)^n \arrow[r] & \mathcal{T}(p) \end{tikzcd} \qquad \begin{tikzcd} \mathcal{T}(n) \times \mathcal{T}(m)^k \arrow[r] \arrow[d] & \mathcal{T}(n) \times \mathcal{T}(m)^n \arrow[d] \\ \mathcal{T}(k) \times \mathcal{T}(m)^k \arrow[r] & \mathcal{T}(m) \end{tikzcd}$$

commute. (Sometimes the right hand diagram is called an extraordinary naturality.)

That concludes my preferred category theoretic definition of algebraic theory. It is easy to see that with the definitions sketched above each cartesian operad gives rise to an algebraic theory. Conversely given an algebraic theory $\mathcal{T}$ one can recover the corresponding cartesian operad in a simple fashion. In order to define the operadic composition

$$\mathcal{T}(n) \times (\mathcal{T}(m_1) \times \cdots \mathcal{T}(m_n)) \longrightarrow \mathcal{T}(m_1 + \cdots + m_n)\,,$$

let $\mathbf{in}_i : m_i \to \Sigma\mathbf{m}$ be the standard injections into the coproduct. Then

$$f[g_1, \cdots, g_n] = f(\mathbf{in}_{1*}g_1, \cdots, \mathbf{in}_{n*}g_n)\,.$$

It is then straightforward though perhaps a little tedious to establish the axioms for a cartesian operad.

The category theoretic definition of algebraic theory has the merit of being more succinct than the operadic definition while keeping in focus the important aspect of functoriality. Hence I refer to it simply as an algebraic theory. However if one leaves the category theoretic perspective aside then one can make to do with a simple and elementary formulation.

2.3 Abstract clones

There is something close to redundancy in the category theoretic definition of algebraic theory just given and it is worth looking a bit more closely at that. Suppose that we have $f : n \to m$. It is compelling to use the composition to define

$$f_* : \mathcal{T}(n) \to \mathcal{T}(m); t \mapsto t(\mathbf{pr}_{f(1)}, \cdots, \mathbf{pr}_{f(n)})\,.$$

Taking t and $\mathbf{pr}_1, \cdots, \mathbf{pr}_m$ at the top left of the extraordinary naturality diagram

$$\begin{tikzcd} \mathcal{T}(n) \times \mathcal{T}(m)^m \arrow[r] \arrow[d] & \mathcal{T}(m) \times \mathcal{T}(m)^m \arrow[d] \\ \mathcal{T}(n) \times \mathcal{T}(m)^n \arrow[r] & \mathcal{T}(m) \end{tikzcd}$$

shows that $\mathcal{T}(f)(t) = f_*t$ so the (hopefully functorial) action can be defined in terms of the composition.

Now it is perhaps a slightly eccentric feature of the definition of algebraic theory that we do not require that the projections project in the sense that $\mathbf{pr}_k(c_1, \cdots, c_n) = c_k$. But suppose we have that. Then for $f : n \to m$ and $g : m \to p$

$$\begin{aligned} g_*(f_*t) &= t(\mathbf{pr}_{f(1)}, \cdots, \mathbf{pr}_{f(n)})(\mathbf{pr}_{g(1)}, \cdots, \mathbf{pr}_{g(m)}) \\ &= t(\mathbf{pr}_{f(1)}(\mathbf{pr}_{g(1)}, \cdots, \mathbf{pr}_{g(m)}), \cdots, \mathbf{pr}_{f(n)}(\mathbf{pr}_{g(1)}, \cdots, \mathbf{pr}_{g(m)})) \\ &= t(\mathbf{pr}_{gf(1)}, \cdots, \mathbf{pr}_{gf(n)}) \\ &= (gf)_*t\,. \end{aligned}$$

Since we already have $\mathbf{id}_*t = t(\mathbf{id}) = t$ we get functoriality of the action defined in terms of composition. With that definition the simple naturality condition is trivial and the extraordinary naturality follows by an obvious argument parallel to that just given.

Why did we not need to require that projections project? Well arguing in the opposite direction it is easy to see that the projection equation follows from extraordinary naturality. Take the square

$$\begin{tikzcd} \mathcal{T}(1) \times \mathcal{T}(m)^n \arrow[r] \arrow[d] & \mathcal{T}(1) \times \mathcal{T}(m) \arrow[d] \\ \mathcal{T}(n) \times \mathcal{T}(m)^n \arrow[r] & \mathcal{T}(m) \end{tikzcd}$$

for the map $k : 1 \to n$. Start with $\mathbf{id}$ together with $(c_1, \cdots\ c_n)$ in $\mathcal{T}(1) \times \mathcal{T}(m)^n$. Going round the top takes us first to $\mathbf{id}$ and c_k and so to c_k. On the other hand going round the bottom takes us first to $k_*\mathbf{id} = \mathbf{pr}_k$ and $(c_1, \cdots\ c_n)$ and so to $\mathbf{pr}_k(c_1, \cdots\ c_n)$. Thus $\mathbf{pr}_k(c_1, \cdots\ c_n) = c_k$ and the projections do indeed act as projections. It follows from this discussion that we can give an alternative more concrete formulation of the notion of algebraic theory as follows.

Definition 2.3 (Universal algebra version) An *abstract clone* is given by the following data.

- For each natural number n a set $\mathcal{T}(n)$.
- Fora each n, elements $\mathbf{pr}_k \in \mathcal{T}(n)$, $1 \leq k \leq n$.
- For each n and m a composition

$$\mathcal{T}(n) \times \mathcal{T}(m)^n \to \mathcal{T}(m); \quad (f, g_1, \cdots, g_n) \mapsto f(g_1, \cdots, g_n)\,.$$

This data is required to satisfy the following equations.

- *Unit laws*

$$\mathbf{pr}_k(a_1, \cdots, a_n) = a_k\,; \qquad a(\mathbf{pr}_1, \cdots, \mathbf{pr}_k) = a\,.$$

- *Associative law*

$$a\big(b_1(c_1,\cdots c_m),\cdots,b_n(c_1,\cdots c_m)\big)=\big(a(b_1,\cdots,b_n)\big)(c_1,\cdots c_m)\,.$$

That is a basic straightforward definition of algebraic theory in a syntactic spirit coming from universal algebra. It is commonly associated with Philip Hall. The attribution is plausible but the usual references do not in fact support the connection and I suppose that it is based on personal knowledge. The terminology may well be due to Taylor [12]. Often in universal algebra one omits the the explicit constants $\mathcal{T}(0)$ in an abstract clone. For the general notion of clone the question of nullary operations is discussed in a paper of Mike Behrisch. Leaving out nullary operations feels like an oversight but curiously the monoid derived from a Λ-algebra is best seen as a fragment of an abstract clone without (explicit) nullary operations.

2.4 Remarks on equivalence

I believe that I have said enough to make it clear that the three formulations of the notion of algebraic theory as cartesian operad, algebraic theory and abstract clone are equivalent. Explicitly in each of the formulations we have a natural straightforward notion of *map* $F:\mathcal{S}\to\mathcal{T}$ *of algebraic theories* given by a (natural) transformation with components $F_n:\mathcal{S}(n)\to\mathcal{T}(n)$ preserving projections and composition. So in each case we get a category of algebraic theories and the translations sketched above give isomorphisms between these categories.

3 Lambda Calculus

I set the rest of this paper in context with a brief review of the approach to the semantics of the λ-calculus given in [5]. In that paper I preserved as much as I could of traditional terminology but I felt it unavoidable to part from tradition in one respect. I propose calling my basic semantic notion of an interpretation of the lambda calculus a *lambda theory* or *λ-theory*. The established use of the term is for a theory in the language of the pure λ-calculus: such a theory corresponds to a quotient of the initial λ-theory in the sense which I favour. I make this change to a far more inclusive sense as a way of insisting on a change of perspective away from the largely syntactic preoccupations of the past.

3.1 Lambda theories and algebras

In [5] I make the case that the need to handle variable binding explicitly in any clean treatment of the λ-calculus means that the basic semantic analysis of the calculus should be in term of algebraic theories. To interpret the λ-calculus I propose λ-theories, which are algebraic theories equipped with further data. Here is the definition.

Definition 3.1 A *λ-theory* is an algebraic theory $\mathcal{L}$ equipped with semi-closed structure. To equip an algebraic theory $\mathcal{L}$ with *semi-closed structure* is to give

a retraction $\mathcal{L}(n+1) \lhd \mathcal{L}(n)$, where the maps

$$\rho : \mathcal{L}(n) \to \mathcal{L}(n+1) \quad \text{and} \quad \lambda : \mathcal{L}(n+1) \to \mathcal{L}(n)$$

are natural in n, and moreover are compatible with the actions

$$\mathcal{L}(m) \times \mathcal{L}(n)^m \to \mathcal{L}(n) \quad \text{and} \quad \mathcal{L}(m+1) \times \mathcal{L}(n)^m \to \mathcal{L}(n+1)\,.$$

The traditional approach [2] to semantics uses the difficult technical notion of λ-algebra. In my approach there is something more accessible which exactly corresponds to that. If $\mathcal{L}$ and $\mathcal{M}$ are λ-theories, then a *map* $\mathcal{L} \to \mathcal{M}$ *of* λ*-theories* is a map of algebraic theories which commutes with the retractions ρ and sections λ. This gives a category of λ-theories. The syntax of the λ-calculus modulo β-equality presents an initial object Λ of this category, the initial λ-theory and that leads naturally enough to the following.

Definition 3.2 A Λ-*algebra* is an algebra for the initial λ-theory Λ.

This definition is cleaner than the traditional one of a λ-algebra in that it avoids the explicit interpretation of abstraction. However to avoid misunderstanding I should stress that it is no more practical: it is just as hard to show directly that one has a Λ-algebra as it is to show that one has a λ-algebra. Rather than concentrating on the algebras one should regard the general situation in the following way. The initial algebra $\mathcal{L}(0)$ for any λ-theory $\mathcal{L}$ can be regarded as a Λ-algebra via the unique map $\Lambda \to \mathcal{L}$. For the most part one finds Λ-algebras by finding λ-theories $\mathcal{L}$ and identifying the putative Λ-algebras with $\mathcal{L}(0)$.

The point of view which I recommend is explained by what I call the Fundamental Theorem of the λ-calculus.

Theorem 3.3 (Fundamental Theorem of the λ-Calculus) *There is an equivalence between the category of* λ*-theories and the category of* Λ*-algebras in which a* λ*-theory* $\mathcal{L}$ *corresponds to the* Λ*-algebra* $\mathcal{L}(0)$.

In [5] I give a more refined formulation of this theorem but I shall not be concerned with that here. In the rest of the paper I shall outline those parts of the proof which depend on the induced monoid and present them in terms of a provisional new notion of Λ-monoid.

3.2 Motivation

My approach to the new notion of Λ-monoid is motivated by a desire to clean up the bare hands proof of the Fundamental Theorem of Lambda Calculus from [5]. The essence of Fundamental Theorem is that the whole of a λ-theory $\mathcal{L}$ is encoded in $\mathcal{L}(0)$ considered as a Λ-algebra. Perhaps the point to appreciate about the proof in [5] is that along the way one is led to look first at how the structure of $\mathcal{L}$ is encoded in the structure in $\mathcal{L}(n)$ for other small n, in particular for n equal to 1 and 2. I explain how that comes about.

The main line of argument from [5] with which we are concerned goes as follows. We start with a Λ-algebra A. We are eventually going to identify A with the initial algebra $\mathcal{U}_A(0)$ for a λ-theory $\mathcal{U}_A$. We get the λ-theory $\mathcal{U}_A$ from the universal object U in a presheaf category $PA = PM_A$ for a monoid M_A derived from A. The eventual identification will give an identification of M_A with the monoid $\mathcal{U}_A(1)$ of the λ-theory. The main point is that we have data on the monoid M_A which presents U as a reflexive object in the presheaf category PA. The proof in [5] depends critically on identifying the function space U^U with a presheaf whose underlying subset of M_A which will eventually be identified with $\mathcal{U}_A(2)$. Close inspection shows that all the data is being controlled by what is in fact a truncation of the λ-theory $\mathcal{U}_A$ to the two objects 1 and 2 of $\mathbb{F}$.

3.3 Truncated clones

Let us step back and consider for a moment the truncation of an algebraic theory $\mathcal{T}$ to the objects 1 and 2. What we want is best dealt with in the abstract clone formulation. So we have two sets $\mathcal{T}(1)$ and $\mathcal{T}(2)$. We have the identity $\mathbf{I} \in \mathcal{T}(1)$ and first and second projections $\mathbf{T}, \mathbf{F} \in \mathcal{T}(2)$ where for simplicity I already use notation coming from the λ-calculus. Finally we have four compositions

$$\mathcal{T}(1) \times \mathcal{T}(1) \longrightarrow \mathcal{T}(1); \qquad \mathcal{T}(1) \times \mathcal{T}(2) \longrightarrow \mathcal{T}(2);$$

$$\mathcal{T}(2) \times \mathcal{T}(1)^2 \longrightarrow \mathcal{T}(1): \qquad \mathcal{T}(2) \times \mathcal{T}(2)^2 \longrightarrow \mathcal{T}(2)\,.$$

All this data will satisfy the abstract clone versions of unit and associativity laws for this restricted collection of data. I hope that it will be helpful to indicate what these are in terms of the customary syntax. The unit laws are of two kinds. We have the left rules

$$\begin{array}{cc} \mathbf{I}(a(x)) = a(x) & \mathbf{I}(c(x,y)) = c(x,y) \\ \mathbf{T}(a(x), b(x)) = a(x) & \mathbf{T}(c(x,y), d(x,y)) = c(x,y) \\ \mathbf{F}(a(x), b(x)) = b(x) & \mathbf{F}(c(x,y), d(x,y)) = d(x,y) \end{array}$$

and the right rules

$$a(\mathbf{I}(x) = a(x) \quad c(\mathbf{T}(x,y), \mathbf{F}(x,y)) = c(x,y)\,.$$

The associativity laws are in some sense absorbed in the syntax. There are four such which very simply mean that the syntactic expressions

$$a(b(c(x)))\,, \quad a(b(c(x), d(x))\,, \quad a(b(c(x,y))\,, \quad a(b(c(x,y), d(x,y)))$$

are unambiguous. And there are four concerning the expressions

$$a(b(d(x)), c(d(x)))\,, \quad a(b(d(x,y)), c(d(x,y)))\,,$$

$$a(b(d(x), e(x)), c(d(x), e(x)))\,, \quad a(b(d(x,y), e(x,y)), c(d(x,y), e(x,y)))$$

where there are repeated subterms and the point is that these could have resulted from substitution of the terms for repeated variables. I hope that this is completely

intuitive but if not it may help to see all these equations rewritten in coded form for a Λ-monoid. I present that at the end of Section 3.5.

3.4 The truncation of a λ-theory

Consider now the truncation of a λ-theory $\mathcal{L}$ to the objects 1 and 2. Then naturally we get the data above but also some additional features are apparent. First by the definition of λ-theory we shall have a retraction $\mathcal{L}(2) \triangleleft \mathcal{L}(1)$. Moreover the image of this retraction is determined as a subset of the monoid $\mathcal{L}(1)$. If we write $\circ$ for the basic composition on $\mathcal{L}(1)$ and $\mathbf{1}$ for the crucial interpretation of the λ-term $\lambda x, y.xy$ then we can identify the image as $\{a \in \mathcal{L}(1) | \mathbf{1} \circ a = a\}$. This has a number of simple consequences. First and obviously we can identify $\mathbf{T}$ and $\mathbf{F}$ as elements of $\mathcal{L}(2)$ and so of $\mathcal{L}(1)$. Secondly, by composing with the retraction as needed we can regard all the compositions above as operations of the appropriate arity on all of $\mathcal{L}(1)$. The unit laws get modified just a little as a result but the associativity laws continue to hold just as they are.

Thus far we see that for a λ-theory $\mathcal{L}$, its truncation can be coded into the monoid $\mathcal{L}(1)$. A λ-theory gives a bit more, a special equation which reflects the way the λ-calculus handles functions of many variables and alongside that some extra data which (in a weak sense) codes products. To prepare for the definition I give the extra data here. In syntactic form we shall we have $p, q \in \mathcal{L}(1)$ and $m \in \mathcal{L}(2)$ such that the equations

$$p(m(x, y)) = x = \mathbf{T}(x, y) \qquad q(m(x, y)) = y = \mathbf{F}(x, y)$$

hold. There is a sense in which this can be made to tie down the λ-theory $\mathcal{L}$, though I shall not give the full story in this paper.

3.5 Lambda monoids: coding the algebraic theory

The idea behind the definition of a λ-monoid is that it is a monoid M with structure which serves to represent a truncated algebraic theory on sets $M(1)$ of unary operations and $M(2)$ of binary operations. Motivated by the discussion in the previous sections we start with the monoid $M = M(1)$, giving us an identity $\mathbf{I} \in M$ and composition $\circ : M^2 \to M$ together with equations which we come to in a moment. Then within M we take an idempotent $\mathbf{1}$ and we set $M(2) = \{a \in M \mid \mathbf{1} \circ a = a\}$. So we take further elements $\mathbf{T}$ and $\mathbf{F}$ to represent the projections. These should lie in $M(2)$. Together with the idempotence of $\mathbf{1}$ that gives us equations

$$\mathbf{1} \circ \mathbf{1} = \mathbf{1} \qquad \mathbf{1} \circ \mathbf{T} = \mathbf{T} \qquad \mathbf{1} \circ \mathbf{F} = \mathbf{F}\,.$$

We have already mentioned the monoid composition and we want three further compositions. I introduce notation for these alongside their syntactic interpretation. For consistency first the monoid composition itself is

$$\circ : M \times M \to M\,; \quad (a, b) \mapsto a \circ b\,,$$

with the syntactic interpretation $a(b(x))$. Then there is a composition

$$\circ : M \times M^2 \to M\,; \quad (a,(b,c)) \mapsto a \circ (b,c)\,,$$

with syntactic interpretation $a(b(x), c(x))$. Of course in principle this is a composition $M(2) \times M^2 \to M$ extended as described above. I use the same symbol here because the output of the composition is not restricted. There is a further composition

$$\star : M \times M \to M\,; \quad (a,b) \mapsto a \star b\,,$$

with syntactic interpretation $a(b(x,y))$. Here in principle the composition is of the form $M \times M(2) \to M(2)$ and the notation $\star$ draws attention to the restricted codomain. Finally there is a composition

$$\star : M \times M^2 \to M\,; \quad (a,(b,c)) \mapsto a \star (b,c)\,,$$

with syntactic interpretation $a(b(x,y), c(x,y))$. Now the composition is in principle of the form $M(2) \times M(2)^2 \to M(2)$ and in the notation the pairing (b,c) draws attention to the first $M(2)$ while the $\star$ again signifies the restricted codomain.

We should avoid stupid distinctions and so we need some laws of hygiene, like those above for $\mathbf{T}$ and $\mathbf{F}$, ensuring good behaviour of our non-standard compositions. I list these. The codomain constraints concern only the $\star$ compositions: they are

$$\mathbf{1} \circ (a \star b) = a \star b\,; \quad \mathbf{1} \circ (a \star (b,c)) = a \star (b,c)\,.$$

The compositions which involve pairing call for constraints on what is applied: these are

$$(\mathbf{1} \circ a) \circ (b,c) = a \circ (b,c)\,; \quad (\mathbf{1} \circ a) \star (b,c) = a \star (b,c)\,.$$

Finally one would expect domain constraints:

$$a \star b = a \star (\mathbf{1} \circ b)\,; \quad a \star (b,c) = a \star (\mathbf{1} \circ b, \mathbf{1} \circ c)\,.$$

There is no harm in giving these but in fact they will follow from equations still to come. I shall pay no further attention to questions of redundancy in this paper.

We have our identity, projections and compositions and we need unit and associativity laws. I held off giving these for the monoid so as to handle them all together. Let me give the unit laws first augmenting those for the monoid. These involve the occasional tweaking to take account of the retraction. I present them so they parallel the laws as they appeared in section 3.3. So the left laws are

$$\mathbf{I} \circ a = a \quad \mathbf{I} \star a = \mathbf{1} \circ a$$

$$\mathbf{T} \circ (a,b) = a \quad \mathbf{T} \star (c,d) = \mathbf{1} \circ c$$
$$\mathbf{F} \circ (a,b) = b \quad \mathbf{F} \star (c,d) = \mathbf{1} \circ d$$

and the right laws are

$$a \circ \mathbf{I} = a \quad c \star (\mathbf{T}, \mathbf{F}) = \mathbf{1} \circ c\,.$$

Of course amongst these unit laws are those for the monoid.

I now turn to the associativity laws. Recall the associativity condition from section 2.2 that for all n, m and p the diagram

$$\begin{tikzcd}
\mathcal{T}(n) \times \mathcal{T}(m)^n \times \mathcal{T}(p)^m \arrow[rr] \arrow[d] & & \mathcal{T}(m) \times \mathcal{T}(p)^m \arrow[d] \\
\mathcal{T}(n) \times (\mathcal{T}(m) \times \mathcal{T}(p)^m)^n \arrow[r] & \mathcal{T}(n) \times \mathcal{T}(p)^n \arrow[r] & \mathcal{T}(p)\,,
\end{tikzcd}$$

commutes. In the Λ-Monoid setting we can set n, m and p to be 1 or 2 so we get eight associativity laws corresponding to the eight syntactic forms in section 3.3. I list these in order parallel to that above, together in each case with the domain from the top left hand corner of the diagram.

$$\mathcal{T}(1) \times \mathcal{T}(1) \times \mathcal{T}(1) \qquad (a \circ b) \circ c = a \circ (b \circ c)$$

$$\begin{aligned}
\mathcal{T}(1) \times \mathcal{T}(2) \times \mathcal{T}(1)^2 &\qquad (a \star b) \circ (c, d) = a \circ (b \circ (c, d)) \\
\mathcal{T}(1) \times \mathcal{T}(1) \times \mathcal{T}(2) &\qquad (a \circ b) \star c = a \star (b \star c) \\
\mathcal{T}(2) \times \mathcal{T}(1)^2 \times \mathcal{T}(1) &\qquad (a \circ (b, c)) \circ d = a \circ (b \circ d, c \circ d) \\
\mathcal{T}(1) \times \mathcal{T}(2) \times \mathcal{T}(2)^2 &\qquad (a \star b) \star (c, d) = a \star (b \star (c, d)) \\
\mathcal{T}(2) \times \mathcal{T}(1)^2 \times \mathcal{T}(2) &\qquad (a \circ (b, c)) \star d = a \star (b \star d, c \star d) \\
\mathcal{T}(2) \times \mathcal{T}(2)^2 \times \mathcal{T}(1)^2 &\qquad a \star (b, c)) \circ (d, e) = a \circ (b \circ (d, e), c \circ (d, e)) \\
\mathcal{T}(2) \times \mathcal{T}(2)^2 \times \mathcal{T}(2)^2 &\qquad (a \star (b, c)) \star (d, e) = a \star (b \star (d, e), c \star (d, e))
\end{aligned}$$

Of course the first associativity listed is just that for the monoid.

3.6 Λ-monoids: the λ-calculus aspects

So far there is not much sign of the λ-calculus in what we have put together. Indeed we could very well have $\mathbf{I} = \mathbf{1}$ in which case $M = M(2)$ and then we could be coding any truncated algebraic theory with an isomorphism between $\mathcal{T}(1)$ and $\mathcal{T}(2)$.

The first aspect of the λ-calculus encasulates as we shall see the crucial representation of the function space. Our compositions are not quite independent and in particular we shall want the following special equation relating two of them:

$$a \circ (b \circ c, d) = (a \circ b) \circ (c, d)\,.$$

This equation should be something of a relief. Up to now everything regarding pairs of arguments has been completely symmetric. But the special equation is not symmetric and its twin under the symmetry does not hold.

Finally we have a feature which in some sense internalises the projections. We additional constants $p, q \in M$ and $m \in M$ (intuitively in $M(2)$) with the equations

$$\mathbf{1} \circ m = m \qquad p \star m = \mathbf{T} \qquad q \star m = \mathbf{F}\,.$$

These constants play a crucial role in the proof of the Fundamental Theorem. The way to understand them is to think of m as in $M(2)$ and so as a binary function. It is a weak form of pairing and p, q in $M = M(1)$ are the corresponding unary projections. It is with these functions that we begin to see the subtleties considered by Koymans [7] for his notion of λ-monoid.

3.7 Λ-algebras to Λ-monoids

Suppose that A is a Λ-algebra in the sense of [5], or (what amounts to the same thing) a λ-algebra in the traditional sense [2]. Then one can give the set

$$A(1) = \{a \in A | \mathbf{1}a = a|\}$$

the structure of a λ-monoid as follows. The constants are

$$\mathbf{I} = \lambda x.x, \quad \mathbf{1} = \lambda xy.xy \quad \mathbf{T} = \lambda xy.x, \quad \mathbf{F} = \lambda xy.y,$$

$$p = \lambda x.x\mathbf{T}, \quad q = \lambda x.x\mathbf{F}, \quad m = \lambda yzx.x(yz)$$

and the binary operations are

$$a \circ b = \lambda x.a(bx), \quad a \star b = \lambda xy.a(bxy),$$

$$c \circ (a, b) = \lambda x.c(ax)(bx), \quad c \star (a, b) = \lambda xy.c(axy)(bxy)\,.$$

Proposition 3.4 *Suppose the A is a Λ-algebra. Then $A(1) = \{a \in A|\mathbf{1}a = a\}$ equipped with the structure above is a Λ-monoid.*

Proof. There are a lot of equations to check but it is all a matter of straightforward calculations with λ-terms. I give a couple of illustrative examples. The crucial special equation holds by the following.

$$\begin{aligned} a \circ (b \circ c, d) &= \lambda x.a((b \circ c)x)(dx) = \lambda x.a(b(cx))(dx) \\ &= \lambda x.(a \circ b)(cx)(dx) = (a \circ b) \circ (c, d) \end{aligned}$$

The basic equation for p follows from the calculation

$$\begin{aligned} p \star m &= \lambda xy.p(mxy) = \lambda xy.(\lambda z.z\mathbf{T})mxy \\ &= \lambda xy.mxy\mathbf{T} = \lambda xy.(\lambda uvw.wuv)xy\mathbf{T} \\ &= \lambda xy.\mathbf{T}xy = \lambda xy.x = \mathbf{T} \end{aligned}$$

The calculation for $q \star m$ is very similar. □

3.8 The Reflexive object

Let M be a monoid and consider the presheaf category PM of sets X with a right action $X \times M \to X$ by M. The universal object U is the representable functor,

that is, M itself with the evident action by composition. Suppose now that M has the structure of a λ-monoid. The first aim is to use that data to exhibit U as a reflexive object. Take $M(2) = \{a \in M | \mathbf{1} \circ a = a\}$ with its evident right action by composition. Since $\mathbf{1} \in M$ is idempotent, this is obviously a retract of U in PM. The idea then is that the Λ-monoid structure gives an identification with the function space U^U.

For any $X, Y \in PM$, standard category theory identifies the function space Y^X with the set of M-equivariant maps $\phi : M \times X \to Y$ with right M-action given by $\phi.c(a, x) = \phi(ca, x)$. The evaluation map is

$$Y^X \times X \to Y\,; \quad (\phi, x) \mapsto \phi(I, x)\,.$$

Now consider the special case of U^U when M is a Λ-monoid. Given $c \in M(2)$ we have an evident map $\phi_c : M \times M \to M; (a, b) \mapsto c \circ (a, b)$. and the aim is to show that this provides an identification of $M(2)$ with U^U.

Lemma 3.5 *Any equivariant $\phi : M \times M \to M$ is of the form*

$$(a, b) \mapsto c \circ (a, b)$$

for a unique $c \in M(2)$.

Proof. First note that for any c, the map

$$M \times M \to M\,; \quad (a, b) \mapsto c \circ (a, b)$$

is indeed M-equivariant as

$$(c \circ (a, b)) \circ d = c \circ (a \circ d, b \circ d)$$

by one of the associativity laws. Conversely given any

$$\phi : M \times M \to M\,,$$

we define

$$c_\phi = \phi(p, q) \star m\,.$$

One of our hygiene laws shows $c_\phi \in M(2)$. A simple calculation using an associativity law gives

$$(c \circ (p, q)) \star m = c \circ (p \star m, q \star m) = c \circ (\mathbf{T}, \mathbf{F}) = \mathbf{1} \circ c = c\,,$$

the last equality assuming $c \in M(2)$. Thus one relevant composite is the identity. To show the other we assume that ϕ is equivariant and using another distributive law a few times we get the following calculation

$$\begin{aligned}(\phi(p,q)\star m)\circ(a,b) &= (\phi(p,q)\circ(m\circ(a,b)\\ &= \phi(p\circ(m\circ(a,b),q\circ(m\circ(a,b))\\ &= \phi((p\star m)\circ(a,b),(q\star m)\circ(a,b))\\ &= \phi(\mathbf{T}\circ(a,b),\mathbf{F}\circ(a,b))\\ &= \phi(a,b)\,.\end{aligned}$$

(Equivariance is used for the second line.) Together these calculations give us an isomorphism of sets between $M(2)$ and the equivariant ϕ. □

Of course we don't want a mere isomorphism of sets. Rather we want an isomorphism in PM, that is, an isomorphism of M-sets. For that is what is needed to show that we have a reflexive object in PM.

Proposition 3.6 *If M is a Λ-monoid then the universal object $U \in PM$ acquires the structure of a reflexive object.*

Proof. We show that the isomorphism

$$M(2) \to PM(U\times U, U)\;;\quad (a,b)\mapsto c\circ(a,b)$$

is an isomorphism of M-sets. Action by $d\in M$ is in the first case $c\mapsto c\circ d$ and in the second $\phi\mapsto\phi.d$ where $\phi.d(a,b)=\phi(d\circ a,b)$. But our last symmetry breaking equation is $(c\circ d)\circ(a,b)=c\circ(d\circ a,b)$ which is exactly what is needed. But as an M-set $M(2)$ is a retract of M, so the proposition follows. □

3.9 Recovering a Λ-algebra

Let us step back a moment. In Section 3.7 we saw that a Λ-algebra A gives a Λ-monoid M_A. Then in Section 3.8 we saw that a Λ-monoid M gives a presheaf category PM with a universal object U with explicit reflexive structure $U^U \lhd U$. That was in essence a civilised version of a key argument from [5]. It is more or less a triviality that a reflexive $U^U \lhd U$ in any cartesian category $\mathcal{C}$ gives a λ-theory $\mathcal{U}$ with $\mathcal{U}(n)=\mathcal{C}(U^n,U)$. Finally from any λ-theory $\mathcal{L}$ we get a Λ-algebra $\mathcal{L}(0)$. So in our case we have a Λ-algebra $\mathcal{U}(0)=\mathcal{C}(1,U)$. We can readily identify the Λ-algebra arising from any Λ-monoid. For

$$PM(1,U)=C_M=\{c\in M\,|\,c\circ a=a\ \text{ for all }\ a\in M\}$$

is the set of constants in the monoid. There is the following intuitively plausible characterisation.

Proposition 3.7 *$\mathbf{F}\in C_M$ and $c\in C_M$ if and only if $c=c\circ\mathbf{F}$.*

The proof is not quite trivial and I omit details. I hope to say more about this Λ-algebra on another occasion.

3.10 What is a Λ-Monoid?

In the last section I have set the scene with material either covered in this paper or in [5]. By arguments of the latter, we know that if we start with a Λ-algebra and go round the circle we get a Λ-algebra equipped with a specific natural isomorphism with the original one. The same holds if we start with a λ-theory and go round to find another λ-theory. Again there is a specific natural isomorphism. But what the Λ-monoids sitting in the middle?

The natural question at this point is whether we can identify the Λ-monoid obtained from the induced Λ-algebra C_M with the original Λ-algebra M. Here in the hope of wetting the reader's appetite I merely sketch the position which emerges.

It is straightforward to identify the composition monoid of the Λ-algebra C_M with the monoid $PM(U,U)$ and it is routine category theory to identify that with M. The identification is practically trivial. Each $a \in M$ induces a map $U \to U$ given by $b \mapsto a \circ b$; and each $\alpha : U \to U$ is induced by $\alpha(\mathbf{I})$. So evidently the work is in what happens with the additional structure of a Λ-monoid.

There seems to be a natural order of development. One first checks that the crucial idempotent $\mathbf{1}$ is preserved under the identification. Then one considers the compositions. The most tricky is the composition $a \circ (b,c)$ but direct calculation shows that it is preserved. The fact that the other compositions $a \star b$ and $a \star (b,c)$ are preserved can be read off from distributive laws. (As an aside I note that in the first case it is a law we have already used while in the second the law is new. That might make one pause.) Moving on, the preservation of the constants $\mathbf{T}$ and $\mathbf{F}$ is almost trivial, so it remains to consider the constants p, q and m.

At this point things begin to get interesting. First the $p = \lambda x.x\mathbf{T}$ and $q = \lambda x.x\mathbf{F}$ of the Λ-monoid constructed from C_M can be defined in terms of the original Λ-monoid by

$$p = \mathbf{I} \circ (\mathbf{I}, \mathbf{T} \star \mathbf{F}) \quad \text{and} \quad q = \mathbf{I} \circ (\mathbf{I}, \mathbf{F} \star \mathbf{F})\,.$$

We could obviously take these as definitions, but it seems that we did not have to do so. And what about m? The new $m = \lambda yzx.xyz$ also must have a definition in terms of the original structure, but now one from which it is not obvious how to eliminate the old m. Whatever we decide to do about p and q, it seems that to make the tie up exact one will need m and an equation involving it.

The situation we have arrived at is now very similar to that set out by Koymans in [7]. The equations we are running into are analogues of the various conditions carefully assembled by Koymans when building up his notion of λ-monoid.

4 Closing Remarks

This paper serves to replace the lengthy and unmotivated calculations in the λ-calculus given in [5] by relatively transparent calculations in terms of Λ-monoid structure. I hope that the fresh context will make the arguments more comprehensible at least for some.

However I imagine that the sensitive reader has been left with a sense of disquiet.

Certainly what is achieved does not make the notion of Λ-monoid definitive. As with Koymans [7], it appears that more is needed. It is not clear what position Koymans was inclined to take about his definitions but I believe that there is an unrecognised phenomenon here worth teasing out. I also think that something similar is at play in the established relationship between λ-calculus and combinatory logic and that the two issues illuminate each other. I am putting off the question of what that all means for the moment; but I hope to have made the approach via algebraic theories and the notion of abtsract clone in particular attractive.

References

[1] Adámek, J., Rosický J. and Vitale E. M., " Algebraic Theories", Cambridge Tracts in Mathematics **184**, Cambridge University Press, 2011.

[2] Barendregt, H. P., "The Lambda Calculus: Its Syntax and Semantics", Studies in Logic and the Foundations of Mathematics **103**, Revised Edition, North-Holland, 1984.

[3] Fiore, M., Gambino, N., Hyland, M. and Winskel, G., *Kleisli Bicategories*, in preparation.

[4] Hyland, M. and Power, J. (2007) The Category Theoretic Understanding of Universal Algebra: Lawvere Theories and Monads. In L. Cardelli, M. Fiore and G. Winskel (editors). *Computation, Meaning and Logic. Articles Dedicated to Gordon Plotkin*. Electronic Notes in Theoretical Computer Science **172**, 437-458.

[5] Hyland, J. M. E., *Classical lambda calculus in modern dress*, to appear in Mathematical Structures in Computer Science.

[6] Hyland, J. M. E., *Elements of a theory of algebraic theories*, submitted to Theoretical Computer Science.

[7] Koymans, C. P. J. (1984) *Models of the Lambda Calculus*. PhD Thesis, Utrecht.

[8] Lambek, J and Scott, P. J., "Introduction to higher order categorical logic," Cambridge University Press, 1986.

[9] Lawvere, F. W., "Functorial Semantics of Algebraic Theories," PhD Thesis, Columbia University, 1963. Republished with an introduction in Reprints in Theory and Applications of Categories, **5**, 2004, 1-121.

[10] Scott, D. S. (1980) *Relating Theories of the λ-calculus*, in J. P. Seldin and J. R. Hindley editors, "To H. B. Curry: Essays on Combinatory Logic and Formalism," Academic Press, 1980, 403-450.

[11] Taylor, P., "Recursive Domains, Indexed Category Theory and Polymorphism, " PhD Thesis, University of Cambridge, 1986.

[12] Taylor, W., *Abstract Clone Theory*, in I.G. Rosenberg and G. Sabidussi editors, "Algebras and Orders," Springer, 1993, 507-530.