Models for simply typed lambda-calculi with fixed point combinators and enumerators

Jean-Pierre van Draanen

Acknowledgements

I am grateful to my promotor and supervisor Henk Barendregt and to my co-promotor Marc Bezem for their guidance and support. Further, I would like to thank everyone that has in one way or another contributed to this thesis. Especially I want to express my gratitude to Erik Barendsen for the many times he helped me with LaTeX.

Contents

Introduction

This thesis is about simply typed λ-calculi and about models for simply typed λ-calculi. We also study higher type recursion theories as tools for constructing models. The simply typed λ-calculi that will be studied are extensions of the pure simply typed λ-calculus $\lambda^{\rightarrow}$ by means of δ-reduction. An example of such a λ-calculus is the system $\lambda\mathsf{T}$ which is a λ-calculus variant of Gödel's system T of primitive recursive functionals. Another example is the system $\lambda\mathsf{Y}(\mathcal{N})$, which is almost identical to PCF [Plotkin 1977]. The central feature of $\lambda\mathsf{Y}(\mathcal{N})$ is that it contains fixed point combinators.

This thesis introduces and studies the system $\lambda\mathsf{E}(\mathcal{N})$, which seems not to have been studied before. This system is obtained by adjoining enumerators to $\lambda^{\rightarrow}$ by means of δ-reduction. It turns out that $\lambda\mathsf{E}(\mathcal{N})$ is a stronger system than $\lambda\mathsf{Y}(\mathcal{N})$: $\lambda\mathsf{E}(\mathcal{N})$ contains fixed point combinators (and therefore $\lambda\mathsf{Y}(\mathcal{N})$ can be translated into $\lambda\mathsf{E}(\mathcal{N})$), but on the other hand $\lambda\mathsf{Y}(\mathcal{N})$ does not contain enumerators. However, in the area of computability the systems $\lambda\mathsf{Y}(\mathcal{N})$ and $\lambda\mathsf{E}(\mathcal{N})$ are equally strong: both compute exactly the recursive functions.

One of the main results of this thesis is the construction of a model for $\lambda\mathsf{E}(\mathcal{N})$. Because $\lambda\mathsf{Y}(\mathcal{N})$ can be translated into $\lambda\mathsf{E}(\mathcal{N})$, one has to look for a $\lambda\mathsf{E}(\mathcal{N})$-model among the $\lambda\mathsf{Y}(\mathcal{N})$-models. It is known that the type structure **MON**, which is obtained from the flat cpo $\omega_\perp$ by hereditary application of the monotone-function-space operation, is a $\lambda\mathsf{Y}(\mathcal{N})$-model. It will be shown that **MON** is also a $\lambda\mathsf{E}(\mathcal{N})$-model. This will be done by first constructing a recursion theory $\mathcal{R}$ on **MON**, which is a collection of total, representable functionals on **MON**. This recursion theory $\mathcal{R}$ satisfies a First Recursion Theorem and a Second Recursion Theorem. Using the Second Recursion Theorem it then follows that **MON** is a $\lambda\mathsf{E}(\mathcal{N})$-model.

In fact, a more general result is obtained. We will axiomatically introduce the notion of 'recursion theory on a type structure'. A recursion theory $\mathcal{R}$ on a type structure $\mathfrak{M}$ is a collection of total, $\mathfrak{M}$-representable functionals on $\mathfrak{M}$, which has a number of properties, among which an Enumeration Property and a Substitution Property. Every recursion theory satisfies a First Recursion Theorem and a Second Recursion Theorem. If a type structure $\mathfrak{M}$ carries a recursion theory $\mathcal{R}$, then it follows from the First Recursion Theorem for $\mathcal{R}$ that $\mathfrak{M}$ is a $\lambda\mathsf{Y}(\mathcal{N})$-model, and from the Second Recursion Theorem for $\mathcal{R}$ it follows that $\mathfrak{M}$

is a $\lambda\mathsf{E}(\mathcal{N})$-model. Since not every type structure is a $\lambda\mathsf{E}(\mathcal{N})$-model, not every type structure carries a recursion theory. Further, the type structures that carry recursion theories can be characterized in terms of $\lambda\mathsf{E}(\mathcal{N})$.

Motivation

This thesis is the result of an investigation of some connections between simply typed λ-calculi and recursion theories in higher types. The combinators of a simply typed λ-calculus can be viewed as definitions for higher type objects. Thus simply typed λ-calculi yield notions of higher type definability. These notions of definability can be compared with notions of higher type recursiveness.

In the following we make this more precise. In order to do so we give the following definitions. First we define types: 0 is a type called the ground type, and if σ and τ are types, then $\sigma \to \tau$ is also a type; the set of all types is $\mathbb{T}$. A partial type structure is a collection $\mathfrak{A}$ of typed objects together with a partial application. The set of objects of type σ in $\mathfrak{A}$ is $\mathfrak{A}_\sigma$. We will always assume that $\mathfrak{A}_\sigma \cap \mathfrak{A}_\tau = \emptyset$ if $\sigma \neq \tau$. Under application objects of type $\sigma \to \tau$ are partial operations from $\mathfrak{A}_\sigma$ into $\mathfrak{A}_\tau$. If $a \in \mathfrak{A}_{\sigma\to\tau}$ and $b \in \mathfrak{A}_\sigma$, then ab (if defined) is the result of applying a to b. The partial type structure $\mathfrak{A}$ is called total if for all σ, $\tau \in \mathbb{T}$ and all $a \in \mathfrak{A}_{\sigma\to\tau}$ and $b \in \mathfrak{A}_\sigma$ the application ab is defined. A partial type structure that is total is called a type structure. Type structures will usually be denoted by $\mathfrak{M}$ instead of $\mathfrak{A}$.

If $\sigma = \sigma_1 \to \cdots \to \sigma_k \to 0$ $(= \sigma_1 \to (\sigma_2 \to (\cdots \to (\sigma_k \to 0)\ldots)))$, then $(\mathfrak{A})_\sigma = \mathfrak{A}_{\sigma_1} \times \ldots \times \mathfrak{A}_{\sigma_k}$. A partial functional on $\mathfrak{A}$ is a partial mapping $\varphi : (\mathfrak{A})_\sigma \to_p \mathfrak{A}_0$. Every object $a \in \mathfrak{A}_\sigma$ gives rise to a partial functional $a^* : (\mathfrak{A})_\sigma \to_p \mathfrak{A}_0$ in the following way: $a^*(b_1, \ldots, b_k) \simeq ab_1 \ldots b_k$. The functional a^* is said to be $\mathfrak{A}$-represented by a. The partial type structure $\mathfrak{A}$ is called extensional if the mapping $a \mapsto a^*$ is injective.

Through the mapping $a \mapsto a^*$ a recursion theory on $\mathfrak{A}$ yields a set of recursive objects. We aim at studying λ-calculi such that the definable objects in $\mathfrak{A}$ coincide with the recursive objects in $\mathfrak{A}$.

Definition systems.

A simply typed λ-calculus Σ consists of a set $\mathcal{T}(\Sigma)$ of simply typed λ-terms called Σ-terms, together with a notion of reduction on $\mathcal{T}(\Sigma)$. Let Σ be a simply typed λ-calculus. Then $\mathcal{T}_\sigma(\Sigma)$ is the set of all Σ-terms of type σ, $\mathcal{T}(\Sigma)^0$ is the set of all closed Σ-terms or combinators of Σ, and $\mathcal{T}_\sigma(\Sigma)^0 = \mathcal{T}(\Sigma)^0 \cap \mathcal{T}_\sigma(\Sigma)$. For every $\sigma \in \mathbb{T}$, Var_σ is the set of all variables of type σ, and $Var_{\mathbb{T}} = \bigcup\{Var_\sigma | \sigma \in \mathbb{T}\}$ is the set of all variables. A β-system is a simply typed λ-calculus with at least β-reduction.

Let $\mathfrak{A}$ be a partial type structure. The domain of $\mathfrak{A}$

$$|\mathfrak{A}| = \bigcup_{\sigma\in\mathbb{T}} \mathfrak{A}_\sigma$$

is the set of all objects of $\mathfrak{A}$. A valuation in $\mathfrak{A}$ is a map $\rho : Var_{\mathbb{T}} \to |\mathfrak{A}|$ such that $\rho(x) \in \mathfrak{A}_\sigma$ for $x \in Var_\sigma$. $Val(\mathfrak{A})$ is the set of all valuations in $\mathfrak{A}$. A partial Σ-interpretation in $\mathfrak{A}$ is a partial mapping

$$[\![\,]\!] : \mathcal{T}(\Sigma) \times Val(\mathfrak{A}) \to_p |\mathfrak{A}|$$
$$(M,\rho) \mapsto [\![M]\!]_\rho$$

such that the following hold:

- $M \in \mathcal{T}_\sigma(\Sigma)\ \&\ [\![M]\!]_\rho\downarrow \Rightarrow [\![M]\!]_\rho \in \mathfrak{A}_\sigma$
- $M \equiv N \Rightarrow [\![M]\!]_\rho \simeq [\![N]\!]_\rho$
- $[\![x]\!]_\rho \simeq \rho(x)$
- $[\![MN]\!]_\rho \simeq [\![M]\!]_\rho[\![N]\!]_\rho$
- $[\![\lambda x.M]\!]_\rho\downarrow\ \&\ [\![\lambda x.M]\!]_\rho a \simeq [\![M]\!]_{\rho(x:=a)}$
- $\forall x \in FV(M)[\,\rho_1(x) = \rho_2(x)\,] \Rightarrow [\![M]\!]_{\rho_1} \simeq [\![M]\!]_{\rho_2}$.

If $[\![\,]\!]$ is a partial Σ-interpretation in $\mathfrak{A}$, then $(\mathfrak{A}, [\![\,]\!])$ is called a partial Σ-structure. A partial Σ-structure $(\mathfrak{A}, [\![\,]\!])$ is called a partial Σ-algebra if

$$\Sigma \vdash M = N \Rightarrow \forall\rho \in Val(\mathfrak{A})[\,[\![M]\!]_\rho \simeq [\![N]\!]_\rho\,]$$

for all $M, N \in \mathcal{T}(\Sigma)$.

Now we can define the notion of definition system. Let $\mathfrak{A}$ be a partial type structure. A definition system for $\mathfrak{A}$ is a pair

$$\mathcal{D} = (\Sigma, [\![\,]\!])$$

where Σ is a β-system and $[\![\,]\!]$ is a partial Σ-interpretation in $\mathfrak{A}$ such that $(\mathfrak{A}, [\![\,]\!])$ is a partial Σ-algebra.

Let $\mathcal{D} = (\Sigma, [\![\,]\!])$ be a definition system for $\mathfrak{A}$. The Σ-combinators serve as definitions for objects in $\mathfrak{A}$. An object $a \in |\mathfrak{A}|$ is $\mathcal{D}$-definable or definable in $\mathcal{D}$ if there exists $M \in \mathcal{T}(\Sigma)^0$ with $a \simeq [\![M]\!]$. Often we will speak of Σ-definability instead of $\mathcal{D}$-definability, leaving the interpretation implicit. This will happen in case the interpretation is in some way 'standard'.

Note that for $(\Sigma, [\![\,]\!])$ to be a definition system for $\mathfrak{A}$, the structure $(\mathfrak{A}, [\![\,]\!])$ has to be a partial Σ-algebra. This is done in order to have a correlation between Σ and $\mathfrak{A}$, and not just between $\mathcal{T}(\Sigma)$ and $\mathfrak{A}$. This enables us to view certain objects as interpretations of Σ-terms with their properties proven in Σ and thus valid uniformly in all partial Σ-algebras at once. Moreover, if $(\mathfrak{A}, [\![\,]\!])$ is a partial Σ-algebra, reductions and conversions in Σ can be seen as computations that are valid in $\mathfrak{A}$.

Recursion theories on partial type structures.

The recursion theories that are used to construct $\lambda\mathsf{E}(\mathcal{N})$-models, as mentioned before, are special cases of a more general concept of higher type recursion theory. In general a higher type recursion theory is a family of partial functionals on a partial type structure. Below we make this more precise.

Let $\mathfrak{A}$ be an extensional partial type structure with $\omega \subseteq \mathfrak{A}_0$. A recursion theory on $\mathfrak{A}$ is a family $\mathcal{R}$ of partial functionals on $\mathfrak{A}$ with a number of properties. For every $e \in \omega$ and every $\sigma \in \mathbb{T}$ the recursion theory $\mathcal{R}$ contains a partial functional

$$\{e\}^\sigma : (\mathfrak{A})_\sigma \to_p \mathfrak{A}_0.$$

The functionals $\{e\}^\sigma$ are the $\mathcal{R}$-recursive functionals on $\mathfrak{A}$. An object a in $\mathfrak{A}$ is called $\mathcal{R}$-recursive if the associated functional a^* is $\mathcal{R}$-recursive.

Every recursion theory $\mathcal{R}$ has the Enumeration Property and the Parameter Property. The Enumeration Property implies that for every $\sigma \in \mathbb{T}$ there exists an $\mathcal{R}$-recursive $\mathcal{E}_\sigma : \mathfrak{A}_0 \times (\mathfrak{A})_\sigma \to_p \mathfrak{A}_0$ such that

$$\mathcal{E}_\sigma(e, \vec{a}) \simeq \{e\}^\sigma(\vec{a})$$

for all $e \in \omega$ and $\vec{a} \in (\mathfrak{A})_\sigma$. The Parameter Property states that for every $\sigma \in \mathbb{T}$ there exists an $\mathcal{R}$-recursive $S_\sigma : (\mathfrak{A}_0)^2 \to \mathfrak{A}_0$ such that for all $e, a \in \omega$ and $\vec{b} \in (\mathfrak{A})_\sigma$ we have $S_\sigma(e, a) \in \omega$ and

$$\{S_\sigma(e, a)\}^\sigma(\vec{b}) \simeq \{e\}^{0\to\sigma}(a, \vec{b}).$$

For $\sigma \in \mathbb{T}$ define $\mathbb{F}_\sigma(\mathfrak{A}) = \{a^* | a \in \mathfrak{A}_\sigma\}$; then $\mathbb{F}_\sigma(\mathfrak{A})$ is the set of all $\mathfrak{A}$-representable partial functionals $(\mathfrak{A})_\sigma \to_p \mathfrak{A}_0$. An operator on $\mathfrak{A}$ is a partial mapping

$$\Phi : (\mathfrak{A})_\sigma \times \mathbb{F}_{\tau_1}(\mathfrak{A}) \times \ldots \times \mathbb{F}_{\tau_l}(\mathfrak{A}) \to_p \mathfrak{A}_0.$$

With every operator Φ on $\mathfrak{A}$ we associate a partial functional Φ^* on $\mathfrak{A}$ in the following way. Let $\sigma, \tau \in \mathbb{T}$, $\tau = \tau_1 \to \cdots \to \tau_l \to 0$. Let

$$\Phi : (\mathfrak{A})_\sigma \times \mathbb{F}_{\tau_1}(\mathfrak{A}) \times \ldots \times \mathbb{F}_{\tau_l}(\mathfrak{A}) \to_p (\mathfrak{A})_0.$$

Define $\Phi^* : (\mathfrak{A})_\sigma \times (\mathfrak{A})_\tau \to_p \mathfrak{A}_0$ by

$$\Phi^*(\vec{a}, \vec{b}) \simeq \Phi(\vec{a}, b_1^*, \ldots, b_l^*).$$

The operator Φ is called $\mathcal{R}$-recursive if the associated functional Φ^* is $\mathcal{R}$-recursive.

Having a notion of $\mathcal{R}$-recursiveness for operators we can discuss the First Recursion Theorem. Because not every recursion theory satisfies the First Recursion 'Theorem', the property of satisfying the First Recursion 'Theorem' will be called the First Fixed Point Property, and will be written as $\mathbf{FP_1}$. A recursion theory

$\mathcal{R}$ on $\mathfrak{A}$ is said to have $\mathbf{FP_1}$ if for every $\sigma \in \mathbb{T}$ and every $\mathcal{R}$-recursive operator $\Phi : (\mathfrak{A})_\sigma \times \mathbb{F}_\sigma(\mathfrak{A}) \to_p \mathfrak{A}_0$ there exists an $\mathcal{R}$-recursive functional $\varphi \in \mathbb{F}_\sigma(\mathfrak{A})$ such that for all $\vec{a} \in (\mathfrak{A})_\sigma$

$$\varphi(\vec{a}) \simeq \Phi(\vec{a}, \varphi).$$

We believe this to be the best formulation of $\mathbf{FP_1}$ for recursion theories on arbitrary partial type structures. Stronger forms of $\mathbf{FP_1}$, with least fixed points, can be achieved in some cases, with certain partial orderings on the $\mathbb{F}_\sigma(\mathfrak{A})$'s. For example Platek recursion on **HC** satisfies a strong form of $\mathbf{FP_1}$, and also the recursion theory $\mathcal{R}$ on **MON**, which is to be constructed in chapter 4, satisfies a strong form of $\mathbf{FP_1}$.

Closely connected with $\mathbf{FP_1}$ is the Substitution Property or **SP**. Let $\sigma \in \mathbb{T}$ and $\tau_1, \ldots, \tau_l \in \mathbb{T}$. Let

$$\Phi : (\mathfrak{A})_\sigma \times \mathbb{F}_{\tau_1}(\mathfrak{A}) \times \ldots \times \mathbb{F}_{\tau_1}(\mathfrak{A}) \to_p \mathfrak{A}_0$$

and for every $j \in \{1, \ldots, l\}$ let $\varphi_j : (\mathfrak{A})_\sigma \times (\mathfrak{A})_{\tau_j} \to_p \mathfrak{A}_0$. The functional substitution of Φ and $\varphi_1, \ldots, \varphi_l$ is the partial functional $\psi : (\mathfrak{A})_\sigma \to_p \mathfrak{A}_0$ defined by

$$\psi(\vec{a}) \simeq \Phi(\vec{a}, \lambda\vec{b}_1.\varphi_1(\vec{a}, \vec{b}_1), \ldots, \lambda\vec{b}_l.\varphi_l(\vec{a}, \vec{b}_l)).$$

The recursion theory $\mathcal{R}$ is said to have **SP** if the $\mathcal{R}$-recursive functionals are closed under functional substitution.

For reasons that will become apparent in a few moments the recursion theory $\mathcal{R}$ on $\mathfrak{A}$ is said to have the Platek Property, written **PP**, if every $\mathcal{R}$-recursive partial functional on $\mathfrak{A}$ is $\mathfrak{A}$-representable. In a sense the Platek Property is a completeness property for $\mathfrak{A}$ w.r.t. $\mathcal{R}$.

In the following it will be shown that every recursion theory that has **SP** and **PP** also has $\mathbf{FP_1}$.

0.0.1. THEOREM. $\mathbf{SP}\ \&\ \mathbf{PP} \Rightarrow \mathbf{FP_1}$. $\square$

The proof of this theorem uses the Second Recursion Theorem which states that every recursion theory has the Second Fixed Point Property, written $\mathbf{FP_2}$.

0.0.2. THEOREM (Second Recursion Theorem; $\mathbf{FP_2}$).
Let $\sigma \in \mathbb{T}$ and let $\varphi : \mathfrak{A}_0 \times (\mathfrak{A})_\sigma \to_p \mathfrak{A}_0$ be $\mathcal{R}$-recursive. Then there exists $e_0 \in \omega$ such that

$$\{e_0\}^\sigma(\vec{a}) \simeq \varphi(e_0, \vec{a})$$

for all $\vec{a} \in (\mathfrak{A})_\sigma$. $\square$

The proof of this theorem is fairly standard and uses the Parameter Property.

Now we can prove **SP** & **PP** $\Rightarrow$ $\mathbf{FP_1}$. Suppose the recursion theory $\mathcal{R}$ has **SP** and **PP**. Let $\sigma \in \mathbb{T}$ and let $\Phi : (\mathfrak{A})_\sigma \times \mathbb{F}_\sigma(\mathfrak{A}) \to_p \mathfrak{A}_0$ be $\mathcal{R}$-recursive. Define $\varphi : \mathfrak{A}_0 \times (\mathfrak{A})_\sigma \to_p \mathfrak{A}_0$ by

$$\varphi(e, \vec{a}) \simeq \Phi(\vec{a}, \lambda\vec{b}.\mathcal{E}_\sigma(e, \vec{b})).$$

Using **SP** it follows that φ is $\mathcal{R}$-recursive. By $\mathbf{FP_2}$ there exists $e_0 \in \omega$ with

$$\{e_0\}^\sigma(\vec{a}) \simeq \varphi(e_0, \vec{a})$$

for all $\vec{a} \in (\mathfrak{A})_\sigma$. Using **PP** we get that $\{e_0\}^\sigma$ is $\mathfrak{A}$-representable, so $\{e_0\}^\sigma \in \mathbb{F}_\sigma(\mathfrak{A})$, and a simple computation shows that for all $\vec{a} \in (\mathfrak{A})_\sigma$

$$\{e_0\}^\sigma(\vec{a}) \simeq \Phi(\vec{a}, \{e_0\}^\sigma).$$

So $\mathcal{R}$ has $\mathbf{FP_1}$.

Next we take a short look at a few recursion theories that arose in the early days of higher type recursion theory. The subject of higher type recursion theory started with the introduction in [Kleene 1959] of schemas S1-S9 that generate a recursion theory of partial functionals on the total objects of all pure types over the natural numbers. The schemas S1-S9 can be seen as generating a recursion theory of partial functionals on the full type structure $\mathfrak{M}(\omega)$ over ω. Although Kleene's S1-S9 recursion theory is a successful theory of computability in higher types, it has none of the properties $\mathbf{FP_1}$, **SP** and **PP**.

[Platek 1966] introduced the partial type structure **HC** of hereditarily consistent objects over ω, together with a recursion theory on **HC**. The Platek recursion theory has all of the properties $\mathbf{FP_1}$, **SP** and **PP**. Platek recursion theory even has a strong form of $\mathbf{FP_1}$: least fixed points of Platek recursive operators on **HC** are Platek recursive. Platek recursion has **PP** because every Platek recursive functional is hereditarily consistent and therefore **HC**-representable.

Recursive and complete definition systems.

Now we will make the connection between recursion theories and simply typed λ-calculi, or more specifically: definition systems. Let $\mathfrak{A}$ be an extensional partial type structure with $\omega \subseteq \mathfrak{A}_0$ and let $\mathcal{R}$ be a recursion theory on $\mathfrak{A}$. Let $\mathcal{D} = (\Sigma, [\![\,]\!])$ be a definition system for $\mathfrak{A}$. Then $\mathcal{D}$ is called $\mathcal{R}$-recursive if every $\mathcal{D}$-definable object of $\mathfrak{A}$ is $\mathcal{R}$-recursive. If $\mathcal{D}$ is $\mathcal{R}$-recursive, then Σ-combinators are definitions for $\mathcal{R}$-recursive objects, and sometimes constructions of $\mathcal{R}$-recursive objects can be carried out in Σ. An $\mathcal{R}$-recursive definition system $\mathcal{D}$ for $\mathfrak{A}$ is called $\mathcal{R}$-complete, or complete for $\mathcal{R}$, if every $\mathcal{R}$-recursive object in $\mathfrak{A}$ is $\mathcal{D}$-definable.

If $\mathcal{D} = (\Sigma, [\![\,]\!])$ is an $\mathcal{R}$-complete definition system for $\mathfrak{A}$, and if moreover $\mathcal{R}$ has **PP**, which is a completeness property as well, then the Σ-combinators also form a complete set of definitions for the $\mathcal{R}$-recursive functionals through the mapping $a \mapsto a^*$. If this is the case $\mathcal{D}$, in a sense, captures all of $\mathcal{R}$.

This raises the following questions:

- Do there exist recursion theories with complete definition systems; preferably recursion theories with **SP** and **PP**?
- Which simply typed λ-calculi are suitable for building definition systems that are complete for some recursion theory?

The next theorem shows that for some recursion theories complete definition systems do not exist. One of the consequences of this theorem is that there is no definition system for **HC** that is complete for Platek recursion.

0.0.3. THEOREM. *Suppose $\mathfrak{A}$ and $\mathcal{R}$ are such that there exists a totally undefined, $\mathcal{R}$-recursive object in $\mathfrak{A}_{0\to 0}$, that is, there exists an $\mathcal{R}$-recursive $a \in \mathfrak{A}_{0\to 0}$ such that $ab \uparrow$ for all $b \in \mathfrak{A}_0$. Then there does not exist an $\mathcal{R}$-complete definition system for $\mathfrak{A}$.*

PROOF. Suppose, towards a contradiction, that $\mathcal{D} = (\Sigma, [\![\ ]\!])$ is an $\mathcal{R}$-complete definition system for $\mathfrak{A}$. By $\mathcal{R}$-completeness there exists $M \in \mathcal{T}_{0\to 0}(\Sigma)^0$ with $[\![M]\!]b \uparrow$ for all $b \in \mathfrak{A}_0$. Because $0 \in \mathfrak{A}_0$ is $\mathcal{R}$-recursive there exists, again by $\mathcal{R}$-completeness, $N \in \mathcal{T}_0(\Sigma)^0$ with $[\![N]\!] \simeq 0$. Then $[\![MN]\!] \simeq [\![M]\!][\![N]\!] \uparrow$, so $[\![(\lambda x.N)(MN)]\!] \simeq [\![\lambda x.N]\!][\![MN]\!] \uparrow$. But $(\lambda x.N)(MN) =_\beta N$ and $(\mathfrak{A}, [\![\ ]\!])$ is a partial Σ-algebra, so $[\![(\lambda x.N)(MN)]\!] \simeq [\![N]\!] \simeq 0$. Contradiction. $\Box$

One way of avoiding situations in which there exist recursive, totally undefined objects is by restricting ourselves to total type structures. This leads to the interesting question if there exist recursion theories with **SP** and **PP** on total type structures. In chapter 4 this question will be answered affirmatively. Let us call a recursion theory $\mathcal{R}$ on a total type structure $\mathfrak{M}$ special if $\mathcal{R}$ has **SP** and **PP**. A type structure $\mathfrak{M}$ is called special if there exists a special recursion theory on $\mathfrak{M}$. The special recursion theories are precisely the recursion theories, mentioned before, that are used to construct $\lambda\mathsf{E}(\mathcal{N})$-models. Except for this introduction we only study special recursion theories in this thesis, so in chapter 4, where we study these recursion theories, we will omit the adjective 'special'. So in chapter 4 by 'recursion theory' we always mean 'special recursion theory'. Note that if $\mathcal{R}$ is a special recursion theory on a type structure $\mathfrak{M}$, then every $\mathcal{R}$-recursive functional must be total. This suggest that $\mathfrak{M}_0$ must contain, next to the natural numbers, objects that simulate undefinedness. The next lemma supports this suggestion.

0.0.4. LEMMA. *If $\mathfrak{A}_0 = \omega$ and $\mathcal{R}$ has* **PP**, *then there exists an $\mathcal{R}$-recursive, totally undefined object in $\mathfrak{A}_{0\to 0}$.*

PROOF. Suppose $\mathfrak{A}_0 = \omega$ and $\mathcal{R}$ has **PP**. Using $\mathfrak{A}_0 = \omega$ and the fact that the successor function $S^+ : \mathfrak{A}_0 \to \mathfrak{A}_0$ is $\mathcal{R}$-recursive, it is easy to show that the totally undefined partial function $\mathfrak{A}_0 \to_p \mathfrak{A}_0$ is $\mathcal{R}$-recursive. By **PP** this function is $\mathfrak{A}$-represented by an $\mathcal{R}$-recursive, totally undefined object in $\mathfrak{A}_{0\to 0}$. $\square$

Because $\mathbf{HC}_0 = \omega$ and Platek recursion has **PP**, it follows that there does not exist a definition system for **HC** that is complete for Platek recursion.

[Platek 1966] gives an index-free characterization of the Platek recursive functionals, that involves a set of combinatory terms that we will call the Platek terms. These Platek terms can be interpreted in **HC** in such a way that the Platek recursive functionals are obtained as the interpretations of the closed Platek terms. The Platek terms provide a connection between $\lambda\mathsf{Y}(\mathcal{N})$ and Platek recursion. Using a translation of $\lambda\mathsf{Y}(\mathcal{N})$ into the Platek terms, one can construct a partial $\lambda\mathsf{Y}(\mathcal{N})$-interpretation $[\![\ ]\!]$ in **HC**. The Platek recursive functionals can now be characterized as the interpretations of the $\lambda\mathsf{Y}(\mathcal{N})$-combinators in **HC**. From this it follows that $(\lambda\mathsf{Y}(\mathcal{N}), [\![\ ]\!])$ can not be a definition system for **HC**, it turns out that $(\mathbf{HC}, [\![\ ]\!])$ is not a partial $\lambda\mathsf{Y}(\mathcal{N})$-algebra. This is caused by the fact that not every $\lambda\mathsf{Y}(\mathcal{N})$-combinator has an interpretation in **HC**. This suggests that $\lambda\mathsf{Y}(\mathcal{N})$ 'defines' objects that cannot be found in **HC**, but that are part of a larger structure. A good candidate for this 'larger structure' is the (total) type structure **MON**. About this type structure we make the following remarks:

- **HC** is embeddable in **MON**.
- There exists a standard $\lambda\mathsf{Y}(\mathcal{N})$-interpretation $[\![\ ]\!]$ in **MON** such that the structure $(\mathbf{MON}, [\![\ ]\!])$ is a $\lambda\mathsf{Y}(\mathcal{N})$-model. So $(\lambda\mathsf{Y}(\mathcal{N}), [\![\ ]\!])$ is a definition system for **MON**. This gives a natural notion of $\lambda\mathsf{Y}(\mathcal{N})$-definability for objects in **MON**.

Trying to improve on the relation between $\lambda\mathsf{Y}(\mathcal{N})$ and Platek recursion on **HC**, we wonder if there exists a special recursion theory $\mathcal{R}$ on **MON** such that the set of all $\mathcal{R}$-recursive objects in **MON** coincides with the set of all $\lambda\mathsf{Y}(\mathcal{N})$-definable objects in **MON**. If such a recursion theory $\mathcal{R}$ exists, then $(\lambda\mathsf{Y}(\mathcal{N}), [\![\ ]\!])$ would be an $\mathcal{R}$-complete definition system for **MON**. Although we have not found a solution to this problem, we did find a partial solution. We have been able to construct a special recursion theory $\mathcal{R}$ on **MON**. Because $\mathcal{R}$ has **SP** and **PP** it also has $\mathbf{FP_1}$. The recursion theory $\mathcal{R}$ even satisfies a strong version of $\mathbf{FP_1}$: least fixed points of $\mathcal{R}$-recursive operators are $\mathcal{R}$-recursive. This implies that every $\lambda\mathsf{Y}(\mathcal{N})$-definable object of **MON** is $\mathcal{R}$-recursive, so $(\lambda\mathsf{Y}(\mathcal{N}), [\![\ ]\!])$ is an $\mathcal{R}$-recursive definition system for **MON**. It is an open problem whether or not every $\mathcal{R}$-recursive object of **MON** is $\lambda\mathsf{Y}(\mathcal{N})$-definable, so we do not know if $(\lambda\mathsf{Y}(\mathcal{N}), [\![\ ]\!])$ is $\mathcal{R}$-complete.

Next we will discuss the position of the recursion theory $\mathcal{R}$ on **MON** relative to Kleene recursion and Platek recursion in view of theorem 0.0.3 and lemma

0.0.4. Together 0.0.3 and 0.0.4 show that given a partial type structure $\mathfrak{A}$ and a recursion theory $\mathcal{R}_1$ on $\mathfrak{A}$, the following combination of three is impossible:

1. $\mathfrak{A}_0 = \omega$;
2. $\mathcal{R}_1$ has the Platek Property;
3. there exists an $\mathcal{R}_1$-complete definition system for $\mathfrak{A}$.

Taking $\mathcal{R}_1$ to be Kleene's S1-S9 recursion theory on $\mathfrak{A} = \mathfrak{M}(\omega)$, points 1 and 3 are satisfied. The existence of a definition system for $\mathfrak{M}(\omega)$ that is complete for Kleene recursion is easily derivable from the following Substitution Theorem, which states that Kleene recursion satisfies a restricted version of **SP**.

0.0.5. THEOREM (Substitution Theorem, [Kleene 1959]). *Let* $\mathfrak{A} = \mathfrak{M}(\omega)$. *Let* $\sigma, \tau \in \mathbb{T}$ *and let* $\varphi : (\mathfrak{A})_\sigma \times \mathfrak{A}_\tau \to \mathfrak{A}_0$ *and* $\psi : (\mathfrak{A})_\sigma \times (\mathfrak{A})_\tau \to \mathfrak{A}_0$ *be total functionals on* $\mathfrak{A}$. *The substitution of* φ *and* ψ *is the functional* $\chi : (\mathfrak{A})_\sigma \to \mathfrak{A}_0$ *defined by*

$$\chi(\vec{a}) = \varphi(\vec{a}, \lambda\vec{b}.\psi(\vec{a}, \vec{b}))$$

for all $\vec{a} \in (\mathfrak{A})_\sigma$. *Here, the expression*

$$\lambda\vec{b}.\psi(\vec{a}, \vec{b})$$

denotes the object in $\mathfrak{A}_\tau$ *that* $\mathfrak{A}$*-represents the functional*

$$\vec{b} \mapsto \psi(\vec{a}, \vec{b}).$$

If φ *and* ψ *are Kleene recursive, then* χ *is also Kleene recursive.* $\square$

0.0.6. COROLLARY. *There exists a definition system for* $\mathfrak{M}(\omega)$ *that is complete for Kleene recursion.*

PROOF. For every Kleene recursive object a in $\mathfrak{M}(\omega)$ take a constant $\mathbf{c}_a$ and build a set $\mathcal{T}(\mathbb{C})$ of λ-terms from these constants. Take $\Sigma = (\mathcal{T}(\mathbb{C}), \beta)$ and let $[\![\,]\!]$ be the Σ-interpretation in $\mathfrak{M}(\omega)$ with $[\![\mathbf{c}_a]\!] = a$. By the extensionality of $\mathfrak{M}(\omega)$ it follows that $(\mathfrak{M}(\omega), [\![\,]\!])$ is a Σ-algebra, so $\mathcal{D} = (\Sigma, [\![\,]\!])$ is a definition system for $\mathfrak{M}(\omega)$.

The fact that $\mathcal{D}$ is Kleene recursive (that is, every $\mathcal{D}$-definable object is Kleene recursive) can be shown using the Substitution Theorem as follows. For all $\sigma, \tau \in \mathbb{T}$ the type structure $\mathfrak{M}(\omega)$ contains an object $\mathbf{k}_{\sigma,\tau}$ of type $\sigma \to \tau \to \sigma$ with $\mathbf{k}_{\sigma,\tau}ab = a$, and for all $\sigma, \tau, \upsilon \in \mathbb{T}$ the type structure $\mathfrak{M}(\omega)$ contains an object $\mathbf{s}_{\sigma,\tau,\upsilon}$ of type $(\sigma \to \tau \to \upsilon) \to (\sigma \to \tau) \to \sigma \to \upsilon$ with $\mathbf{s}_{\sigma,\tau,\upsilon}abc = ac(bc)$.

Let $\mathcal{X}$ be the least set of objects in $\mathfrak{M}(\omega)$ satisfying

(i) $\mathbf{k}_{\sigma,\tau} \in \mathcal{X}$

(ii) $\mathbf{s}_{\sigma,\tau,\upsilon} \in \mathcal{X}$

(iii) $a \in \mathcal{X}$ for every Kleene recursive object a in $\mathfrak{M}(\omega)$
(iv) $\mathcal{X}$ is closed under application.

Then $\mathcal{X}$ consists precisely of all $\mathcal{D}$-definable objects. Using Kleene's Substitution Theorem one can show

- the set of all Kleene recursive objects is closed under application;
- all objects $\mathbf{k}_{\sigma,\tau}$ and $\mathbf{s}_{\sigma,\tau,\upsilon}$ are Kleene recursive.

Therefore $\mathcal{X}$ is contained in the set of all Kleene recursive objects. So $\mathcal{D}$ is Kleene recursive. But clearly every Kleene recursive object is in $\mathcal{X}$, so $\mathcal{D}$ is also complete. □

We see that Kleene recursion satisfies 1 and 3 of the abovementioned impossible combination of 1, 2 and 3. Platek recursion satisfies 1 and 2. The 'missing' combination of 2 and 3 is satisfied by the recursion theory $\mathcal{R}$ on **MON**. (Note that $\mathbf{MON}_0 = \omega_\perp \neq \omega$.)

Having a special recursion theory $\mathcal{R}$ on a total and extensional type structure $\mathfrak{M}$, we can ask ourselves if there exists an $\mathcal{R}$-complete definition system for $\mathfrak{M}$. The answer is affirmative and again the construction is trivial: for every $\mathcal{R}$-recursive object a of $\mathfrak{M}$ take a constant $\mathbf{c}_a$ and build a set $\mathcal{T}(\mathbb{C})$ of λ-terms from these constants, take $\Sigma = (\mathcal{T}(\mathbb{C}), \beta)$, and let $[\![\,]\!]$ be the Σ-interpretation in $\mathfrak{M}$ with $[\![\mathbf{c}_a]\!] = a$; then $(\Sigma, [\![\,]\!])$ is an $\mathcal{R}$-complete definition system for $\mathfrak{M}$. This is unsatisfactory. Instead of looking for complete definition systems in particular cases, it is more interesting to know if there exist simply typed λ-calculi that capture the essence of the notion of recursion theory. In order to make this idea more precise we define the notion of ultra-complete ω-system.

An ω-system consists of a β-system together with a numeral system. If Σ is a β-system, then a numeral system in Σ is a sequence $\mathbf{a} = (\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2, \ldots)$ in $\mathcal{T}_0(\Sigma)^0$ such that

$$\Sigma \vdash \mathbf{a}_m = \mathbf{a}_n \Rightarrow m = n$$

for all $m, n \in \omega$. If $\mathbf{a}$ is a numeral system in the β-system Σ, then the pair $(\Sigma, \mathbf{a})$ is called an ω-system.

An ω-system $(\Sigma, \mathbf{a})$ is ultra-complete if for every Σ-model $(\mathfrak{M}, [\![\,]\!])$ with $\mathfrak{M}$ extensional, $\omega \subseteq \mathfrak{M}_0$ and $[\![\mathbf{a}_m]\!] = m$ for all $m \in \omega$, there exists a special recursion theory $\mathcal{R}$ on $\mathfrak{M}$ such that $(\Sigma, [\![\,]\!])$ is an $\mathcal{R}$-complete definition system for $\mathfrak{M}$.

Because the β-systems $\lambda\mathsf{Y}(\mathcal{N})$ and $\lambda\mathsf{E}(\mathcal{N})$ both have a standard, built-in numeral system $\mathbf{c} = (\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2, \ldots)$ (consisting of constants of type 0), the ω-systems $(\lambda\mathsf{Y}(\mathcal{N}), \mathbf{c})$ and $(\lambda\mathsf{E}(\mathcal{N}), \mathbf{c})$ are identified with $\lambda\mathsf{Y}(\mathcal{N})$ and $\lambda\mathsf{E}(\mathcal{N})$ respectively.

0.0.7. Theorem. *The ω-system $\lambda\mathsf{Y}(\mathcal{N})$ is not ultra-complete.*

Proof. Let $(\mathfrak{M}, [\![\,]\!])$ be the (open) term model of $\lambda\mathsf{Y}\eta(\mathcal{N})$. By theorem 4.1.6 the system $\lambda\mathsf{Y}\eta(\mathcal{N})$ does not have enumerators, and from this it follows that there does not exist a special recursion theory $\mathcal{R}$ on $\mathfrak{M}$ such that $(\lambda\mathsf{Y}(\mathcal{N}), [\![\,]\!])$ is an $\mathcal{R}$-complete definition system for $\mathfrak{M}$. $\square$

Searching for ultra-complete ω-systems, it seems natural, in view of the Enumeration Property for recursion theories and in view of the proof of the previous theorem, to look at simply typed λ-calculi with enumerators. The prime example of a simply typed λ-calculus with enumerators is the system $\lambda\mathsf{E}(\mathcal{N})$. This system turns out to be intimately connected with special recursion theories. This connection is expressed by the following facts that are consequences of two theorems that will be shown in section 4.4:

- $\lambda\mathsf{E}(\mathcal{N})$ is ultra-complete.
- Let $\mathcal{R}$ be a special recursion theory on an extensional type structure $\mathfrak{M}$. Then there exists a $\lambda\mathsf{E}(\mathcal{N})$-interpretation $[\![\,]\!]$ in $\mathfrak{M}$ such that $(\mathfrak{M}, [\![\,]\!])$ is a $\lambda\mathsf{E}(\mathcal{N})$-model and $(\lambda\mathsf{E}(\mathcal{N}), [\![\,]\!])$ is an $\mathcal{R}$-recursive definition system for $\mathfrak{M}$.

Because **MON** is a special type structure it follows by the second of these facts that there exists a $\lambda\mathsf{E}(\mathcal{N})$-interpretation $[\![\,]\!]$ in **MON** such that $(\mathbf{MON}, [\![\,]\!])$ is a $\lambda\mathsf{E}(\mathcal{N})$-model.

Although in the original research simply typed λ-calculi and recursion theories on type structures are equal partners, the presentation of the results in this thesis is organized in such a way that simply typed λ-calculi and their models are the prime objects of study, and recursion theories on type structures serve merely as a tool for constructing models.

Summary

The first three chapters form an introduction to simply typed λ-calculi and their models. Chapter 1 deals with the simply typed λ-calculi themselves, and chapter 2 deals with the model theory of simply typed λ-calculi. In chapter 3 cartesian closed categories are used to construct models for simply typed λ-calculi. Most of the theory in these first three chapters is already known, and in these chapters we only occasionally prove some new results. We include these three introductory chapters for two reasons. The first reason is to make this thesis self-contained. The second reason is that we do not know any references that provide a complete and coherent treatment of all the theory on simply typed λ-calculi that we need in this thesis.

Chapter 4 consists entirely of new results. In chapter 4 we introduce and study the system $\lambda\mathsf{E}(\mathcal{N})$ which contains enumerators. We also introduce recursion theories on type structures and study their relation with the systems $\lambda\mathsf{Y}(\mathcal{N})$ and $\lambda\mathsf{E}(\mathcal{N})$. Using recursion theories we construct an extensional $\lambda\mathsf{E}(\mathcal{N})$-model.

In chapter 5 we study diagrams of type structures. For every type structure $\mathfrak{M}$ the diagram of $\mathfrak{M}$ is a simply typed λ-calculus $\lambda(\mathfrak{M})$ that contains representatives for all objects of $\mathfrak{M}$. Among other things we show that $\lambda(\mathfrak{M})$ has the Church-Rosser Property if and only if $\mathfrak{M}$ is quasi-extensional. We also discuss some type-free variants of $\lambda(\mathfrak{M})$. Except for section 5.2, also chapter 5 consists of new results.

Summary of results

Chapter 1 contains no new results.

Chapter 2

Let $\mathfrak{M}$ be a type structure. In section 2.1 we construct, for every $\sigma \in \mathbb{T}$, a topology θ_σ on $\mathfrak{M}_\sigma$.

0.0.8. THEOREM. *$\mathfrak{M}$ is extensional if and only if θ_σ is a Hausdorff topology for every $\sigma \in \mathbb{T}$.*

0.0.9. THEOREM. *The only implications that hold in general between the notions of algebra, model, quasi-algebra and quasi-model are:*
(i) *model $\Rightarrow$ algebra*
(ii) *quasi-model $\Rightarrow$ quasi-algebra*
(iii) *algebra $\Rightarrow$ quasi-algebra*
(iv) *model $\Rightarrow$ quasi-algebra.*

Chapter 3

In chapter 3 we study type structures in cartesian closed categories. Let $\mathcal{C}$ be a cartesian closed category. Then $\mathbf{TS}(\mathcal{C})$ is the category of all type structures in $\mathcal{C}$. An ordinary type structure is a type structure in the ccc **Set** of sets, and $\mathbf{TS} = \mathbf{TS}(\mathbf{Set})$ is the category of all ordinary type structures. The functor * from $\mathbf{TS}(\mathcal{C})$ into $\mathbf{TS}$ assigns an ordinary type structure $\mathbf{m}^*$ to every type structure $\mathbf{m}$ in $\mathcal{C}$.

0.0.10. THEOREM. *If $\mathcal{C}$ is an extensional category and $\mathbf{m}$ is an extensional type structure in $\mathcal{C}$, then $\mathbf{m}^*$ is extensional.*

A subclass of the class of all type structures in $\mathcal{C}$ is the class of all complex type structures in $\mathcal{C}$. (A type structure in $\mathcal{C}$ is complex if it is complete and extensional.)

0.0.11. THEOREM (Isomorphy Theorem).
If $\mathbf{m}^1$ and $\mathbf{m}^2$ are complex type structures in $\mathcal{C}$ with $\mathbf{m}^1_0 \cong \mathbf{m}^2_0$, then $\mathbf{m}^1 \cong \mathbf{m}^2$.

Every non-empty object a of $\mathcal{C}$ gives rise to a complex type structure $\mathbf{m}_{\mathcal{C}}(a)$ in $\mathcal{C}$.

0.0.12. THEOREM (Representation Theorem).
If $\mathbf{m}$ is a complex type structure in $\mathcal{C}$, then $\mathbf{m} \cong \mathbf{m}_{\mathcal{C}}(\mathbf{m}_0)$.

Mon is the extensional ccc of complete partial orders (cpo's) with monotone functions. Every cpo $\mathbb{D}$ gives rise to an extensional type structure $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D}) = \mathbf{m}_{\mathbf{Mon}}(\mathbb{D})^*$.

For every structure $\mathcal{A}$ we have a simply typed λ-calculus $\lambda\mathsf{Y}\eta(\mathcal{A})$ containing fixed point combinators and a representation of the structure $\mathcal{A}$.

0.0.13. THEOREM. *Let $\mathcal{A}$ be a structure and let $\mathbb{D}$ be a cpo. If $\mathcal{A}$ is $\mathbb{D}$-representable, then $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ is a pre-$\lambda\mathsf{Y}\eta(\mathcal{A})$-model.*

0.0.14. THEOREM. *Let $\mathcal{A} = (\omega, \{0\}, S^+)$. The system $\lambda\mathsf{Y}\eta(\mathcal{A})$ has*
(i) *fixed point combinators $\mathbf{Y}_\sigma$*
(ii) *numerals $\mathbf{c}_m$*
(iii) *test-for-zero*
(iv) *successor.*
The predecessor function $P^- : \omega \to \omega$ is not $\lambda\mathsf{Y}\eta(\mathcal{A})$-computable.

Let $\mathcal{A}$ be a structure with domain A. Then $\mathcal{A}$ is $\mathbb{A}_\perp$-representable, where $\mathbb{A}_\perp$ is the flat cpo over A.

0.0.15. THEOREM. *$\mathfrak{M}_{\mathbf{Mon}}(\mathbb{A}_\perp)$ is a pre-$\lambda\mathsf{Y}\eta(\mathcal{A})$-model.*

$\mathbf{Tot}(A)$ is the substructure of $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{A}_\perp)$ consisting of the total objects of $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{A}_\perp)$.

0.0.16. THEOREM. (i) *$\mathfrak{M}(A)$ can be embedded in $\mathbf{Tot}(A)$.*
(ii) *$\mathbf{Tot}(A)$ is quasi-extensional but not extensional.*
(iii) *$\mathbf{Tot}(A)^E \cong \mathfrak{M}(A)$.*

Chapter 4

0.0.17. THEOREM. *Every β-system that has enumerators also has fixed point combinators.*

0.0.18. THEOREM. *The systems $\lambda\mathsf{Y}(\mathcal{N})$ and $\lambda\mathsf{Y}\eta(\mathcal{N})$ do not have enumerators.*

0.0.19. THEOREM. *Let Σ be a system with enumerators. Let $(\mathfrak{M}, [\![\,]\!])$ be a quasi-Σ-algebra. Then $\mathfrak{M}_0$ is either infinite or a singleton.*

0.0.20. THEOREM. (i) *$\lambda\mathsf{E}(\mathcal{N})$ has fixed point combinators.*
(ii) *$\lambda\mathsf{Y}(\mathcal{N}) \trianglelefteq \lambda\mathsf{E}(\mathcal{N})$.*
(iii) *Every recursive function is $\lambda\mathsf{E}(\mathcal{N})$-computable.*

0.0.21. THEOREM. *There exists a translation from $\lambda\mathsf{E}(\mathcal{N})$ into the type-free λ-calculus.*

In section 4.3 we define the notion of recursion theory on so-called ω-structures. Let $\mathfrak{M}$ be an ω-structure. Then $\omega \subseteq \mathfrak{M}_0$. Let $\mathcal{R}$ be a recursion theory on $\mathfrak{M}$.

0.0.22. THEOREM (First Recursion Theorem).
For every $\sigma \in \mathbb{T}$ there exists an $\mathcal{R}$-recursive object $\mathbf{y}_\sigma \in \mathfrak{M}_{(\sigma\to\sigma)\to\sigma}$ such that for all $f \in \mathfrak{M}_{\sigma\to\sigma}$ we have

$$\mathbf{y}_\sigma f \approx f(\mathbf{y}_\sigma f).$$

0.0.23. THEOREM. *There exists a quasi-$\lambda\mathsf{Y}(\mathcal{N})$-interpretation $[\![\,]\!]$ in $\mathfrak{M}$ with*
(i) *$(\mathfrak{M}, [\![\,]\!])$ is a quasi-$\lambda\mathsf{Y}(\mathcal{N})$-model*
(ii) *every definable object of $(\mathfrak{M}, [\![\,]\!])$ is $\mathcal{R}$-recursive*
(iii) *$[\![\mathbf{c}_m]\!] = m$ for all $m \in \omega$.*

0.0.24. THEOREM (Extension Theorem).
Every recursive function on ω has an $\mathcal{R}$-recursive extension.

0.0.25. THEOREM (Second Recursion Theorem).
Let $\sigma \in \mathbb{T}$ and $\varphi : \mathfrak{M}_0 \times (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ be $\mathcal{R}$-recursive. Then there exists $e \in \mathbf{Ix}(\sigma)$ such that for all $\vec{a} \in (\mathfrak{M})_\sigma$

$$\{e\}^\sigma(\vec{a}) = \varphi(e, \vec{a}).$$

0.0.26. THEOREM. *There exists a quasi-$\lambda\mathsf{E}(\mathcal{N})$-interpretation $[\![\,]\!]$ in $\mathfrak{M}$ with*
(i) *$(\mathfrak{M}, [\![\,]\!])$ is a quasi-$\lambda\mathsf{E}(\mathcal{N})$-model*
(ii) *every definable object of $(\mathfrak{M}, [\![\,]\!])$ is $\mathcal{R}$-recursive*
(iii) *$[\![\mathbf{c}_m]\!] = m$ for all $m \in \omega$.*

In the following $\mathfrak{M}$ is an arbitrary ω-structure.

0.0.27. THEOREM. *Let $[\![\,]\!]$ be a quasi-$\lambda\mathsf{E}(\mathcal{N})$-interpretation in $\mathfrak{M}$ such that $(\mathfrak{M}, [\![\,]\!])$ is a quasi-$\lambda\mathsf{E}(\mathcal{N})$-model with $[\![\mathbf{c}_m]\!] = m$ for all $m \in \omega$. Then there exists a recursion theory $\mathcal{R}$ on $\mathfrak{M}$ such that the $\mathcal{R}$-recursive objects of $\mathfrak{M}$ are precisely the definable objects of $(\mathfrak{M}, [\![\,]\!])$.*

An ω-structure $\mathfrak{M}$ is called special if there exists a recursion theory on $\mathfrak{M}$.

0.0.28. THEOREM. *$\mathfrak{M}$ is special if and only if there exists a quasi-$\lambda\mathsf{E}(\mathcal{N})$-interpretation $[\![\,]\!]$ in $\mathfrak{M}$ such that $(\mathfrak{M}, [\![\,]\!])$ is a quasi-$\lambda\mathsf{E}(\mathcal{N})$-model with $[\![\mathbf{c}_m]\!] = m$ for all $m \in \omega$.*

0.0.29. THEOREM. *The ω-structure* **MON** *carries a recursion theory $\mathcal{R}$ with the property that every $\lambda\mathsf{Y}\eta(\mathcal{N})$-definable object of* **MON** *is $\mathcal{R}$-recursive.*

0.0.30. THEOREM. *There exists a $\lambda\mathsf{E}(\mathcal{N})$-interpretation $[\![\,]\!]$ in* **MON** *such that*
 (i) *$(\mathbf{MON}, [\![\,]\!])$ is a $\lambda\mathsf{E}(\mathcal{N})$-model*
 (ii) *$[\![\mathbf{c}_m]\!] = m$ for all $m \in \omega$.*

Chapter 5

To every type structure $\mathfrak{M}$ we assign a simply typed λ-calculus $\lambda(\mathfrak{M})$ called the diagram of $\mathfrak{M}$. For every object a of $\mathfrak{M}$ we have a constant $\mathbf{C}(a)$. The $\lambda(\mathfrak{M})$-terms are built from these constants. The set of all $\lambda(\mathfrak{M})$-terms is $\mathcal{T}(\mathfrak{M})$. The reduction in $\lambda(\mathfrak{M})$ consists of β-reduction and a form of δ-reduction called μ-reduction.

The diagram $\lambda(\mathfrak{M})$ gives rise to a relation $\rhd$ between closed $\lambda(\mathfrak{M})$-terms and objects of $\mathfrak{M}$. The relation $\rhd$ is called $\lambda(\mathfrak{M})$-computability, and is defined as follows. For $\sigma = \sigma_1 \to \cdots \to \sigma_k \to 0$, $M \in \mathcal{T}_\sigma(\mathfrak{M})^0$, $a \in \mathfrak{M}_\sigma$ we say that M $\lambda(\mathfrak{M})$-computes a (notation $M \rhd a$) if for all $(b_1, \ldots, b_k) \in (\mathfrak{M})_\sigma$

$$M\mathbf{C}(b_1)\ldots\mathbf{C}(b_k) =_{\beta\mu} \mathbf{C}(ab_1\ldots b_k).$$

0.0.31. THEOREM. *$\lambda(\mathfrak{M}) \models$* **CR** *if and only if $\mathfrak{M}$ is quasi-extensional.*

0.0.32. THEOREM. *If $\mathfrak{M}$ is quasi-extensional, then*

$$M \rhd a \;\&\; M \rhd b \Rightarrow a \approx b.$$

0.0.33. THEOREM (Compatibility Theorem).
If $\mathfrak{M}$ is quasi-extensional, then for all $\sigma, \tau \in \mathbb{T}$, $M \in \mathcal{T}_{\sigma\to\tau}(\mathfrak{M})^0$, $N \in \mathcal{T}_\sigma(\mathfrak{M})^0$, $a \in \mathfrak{M}_{\sigma\to\tau}$, $b \in \mathfrak{M}_\sigma$

$$M \rhd a \;\&\; N \rhd b \Rightarrow MN \rhd ab.$$

In the following $\mathfrak{M}$ is a λ-structure. Then $\mathfrak{M}$ is quasi-extensional and there exists a quasi-$\lambda(\mathfrak{M})$-interpretation $[\![\,]\!]$ in $\mathfrak{M}$ with $[\![\mathbf{C}(a)]\!] = a$ for all $a \in |\mathfrak{M}|$.

0.0.34. THEOREM (First Characterization Theorem).
For all $\sigma \in \mathbb{T}$, $M \in \mathcal{T}_\sigma(\mathfrak{M})^0$, $a \in \mathfrak{M}_\sigma$

$$M \rhd a \iff [\![M]\!] \approx a.$$

0.0.35. THEOREM (Second Characterization Theorem).
The reductor μ on $\mathcal{T}(\mathfrak{M})$ consists of all reduction rules

$$\mathbf{C}(a)M_1\ldots M_k \to \mathbf{C}(a[\![M_1]\!]\ldots[\![M_k]\!])$$

with $\sigma = \sigma_1 \to \cdots \to \sigma_k \to 0 \neq 0$, $a \in \mathfrak{M}_\sigma$, $M_i \in \mathcal{T}_{\sigma_i}(\mathfrak{M})^0$ $(i = 1, \ldots, k)$.

0.0.36. THEOREM. *$(\mathfrak{M}, [\![\,]\!])$ is a quasi-$\lambda(\mathfrak{M})$-model.*

Notation

ω is the set $\{0, 1, 2, 3, \ldots\}$ of all natural numbers and **On** is the class of all ordinal numbers.

Let A be a set. The power set of A is $\wp(A) = \{B | B \subseteq A\}$. The cardinality of A is $\mathbf{card}(A)$. The set of all finite sequences of elements of A is $A^* = \bigcup\{A^k | k \in \omega\}$.

Chapter 1

Simply typed lambda-calculi

This chapter is devoted to the syntactic study of simply typed λ-calculi. Simply typed λ-calculi will usually be called systems for short. In this chapter we will define the general notions of system and reduction. We are mainly concerned with so-called $\beta\delta$-systems and $\beta\eta\delta$-systems. In a nutshell one can say that a $\beta\delta$-system is a system which is obtained from the pure simply typed λ-calculus $\lambda^{\rightarrow}$ by adding constants and δ-reduction, and a $\beta\eta\delta$-system is a system consisting of the pure extensional simply typed λ-calculus $\lambda^{\rightarrow}\eta$ plus constants and δ-reduction. An important example of a $\beta\delta$-system is $\lambda\mathsf{Y}(\mathcal{N})$, and an important example of a $\beta\eta\delta$-system is $\lambda\mathsf{Y}\eta(\mathcal{N})$.

We prove a Church-Rosser theorem for $\beta\delta$-reduction and for $\beta\eta\delta$-reduction, which can be applied to the systems $\lambda\mathsf{Y}(\mathcal{N})$ and $\lambda\mathsf{Y}\eta(\mathcal{N})$ thereby proving the **CR**-property for the reduction relation of these systems.

Other topics in this chapter are: computability in systems with numerals, translations between systems, and simply typed combinatory logic.

1.1. Types and λ-terms.

We define a set of types and we define sets of typed λ-terms with constants.

1.1.1. Definition. The set $\mathbb{T}$ of all (simple) types is defined inductively as follows.

(i) $0 \in \mathbb{T}$ (ground type, zero).

(ii) $\sigma, \tau \in \mathbb{T} \Rightarrow (\sigma \rightarrow \tau) \in \mathbb{T}$ (function types). $\Box$

Notation. (i) For types we use σ,τ,υ,σ_0,σ_1,... .

(ii) Outermost parentheses of types are usually not written.

(iii) For the operation $\rightarrow$ we use association to the right, so for example

$$\sigma_1 \rightarrow \sigma_2 \rightarrow \sigma_3 \rightarrow \sigma_4 = (\sigma_1 \rightarrow (\sigma_2 \rightarrow (\sigma_3 \rightarrow \sigma_4))). \Box$$

1.1.2. LEMMA. *For each $\sigma \in \mathbb{T}$ there exist unique $k \in \omega$ and $\sigma_1,\sigma_2,\dots,\sigma_k \in \mathbb{T}$ such that*

$$\sigma = \sigma_1 \to \sigma_2 \to \cdots \to \sigma_k \to 0.$$

PROOF. Easy, by induction on the structure of the types. □

1.1.3. DEFINITION. Let $\sigma = \sigma_1 \to \cdots \to \sigma_k \to 0 \in \mathbb{T}$.
(i) The types $\sigma_1,\dots,\sigma_k$ are called the factors of σ.
(ii) The number of factors of σ is denoted by $fac(\sigma)$, so $fac(\sigma) = k$. □

Now we turn to simply typed λ-terms. These are built from variables and constants by means of application and abstraction.

1.1.4. DEFINITION. (i) For each $\sigma \in \mathbb{T}$

$$Var_\sigma = \{v_0^\sigma, v_1^\sigma, v_2^\sigma, \dots\}$$

is a countably infinite set of variables of type σ. We assume $v_0^\sigma,v_1^\sigma,\dots$ to be mutually distinct, and we also assume $Var_\sigma \cap Var_\tau = \emptyset$ if $\sigma \neq \tau$.
(ii) The set of all typed variables:

$$Var_{\mathbb{T}} = \bigcup_{\sigma \in \mathbb{T}} Var_\sigma. \ \square$$

1.1.5. DEFINITION. (i) A constant family is a family

$$\mathbb{C} = (\mathbb{C}_\sigma)_{\sigma \in \mathbb{T}}$$

such that for all $\sigma, \tau \in \mathbb{T}$

(1) $\mathbb{C}_\sigma$ is a (possibly empty) set

(2) $\sigma \neq \tau \Rightarrow \mathbb{C}_\sigma \cap \mathbb{C}_\tau = \emptyset$.

(ii) The elements of $\mathbb{C}_\sigma$ are called constants of type σ.
(iii) If $\mathbb{C} = (\mathbb{C}_\sigma)_{\sigma \in \mathbb{T}}$ is a constant family, then

$$|\mathbb{C}| = \bigcup_{\sigma \in \mathbb{T}} \mathbb{C}_\sigma. \ \square$$

NOTATION. (i) For variables we will use $a,b,c,\dots,x,y,z$.
(ii) For constants we will use $\mathbf{C},\mathbf{D},\mathbf{C}_0,\mathbf{C}_1,\dots$ □

1.1.6. DEFINITION. Let $\mathbb{C} = (\mathbb{C}_\sigma)_{\sigma \in \mathbb{T}}$ be a constant family.
(i) For every $\sigma \in \mathbb{T}$ the set $\mathcal{T}_\sigma(\mathbb{C})$ of all λ-terms of type σ with constants from $\mathbb{C}$ is defined by simultaneous induction as follows.

(1) $Var_\sigma \subseteq \mathcal{T}_\sigma(\mathbb{C})$.

(2) $\mathbb{C}_\sigma \subseteq \mathcal{T}_\sigma(\mathbb{C})$.

(3) $M \in \mathcal{T}_{\sigma\to\tau}(\mathbb{C})$ & $N \in \mathcal{T}_\sigma(\mathbb{C}) \Rightarrow (MN) \in \mathcal{T}_\tau(\mathbb{C})$.

(4) $x \in Var_\sigma$ & $M \in \mathcal{T}_\tau(\mathbb{C}) \Rightarrow (\lambda x.M) \in \mathcal{T}_{\sigma\to\tau}(\mathbb{C})$.

(ii) The set of all (simply) typed λ-terms with constants from $\mathbb{C}$:

$$\mathcal{T}(\mathbb{C}) = \bigcup_{\sigma\in\mathbb{T}} \mathcal{T}_\sigma(\mathbb{C}).$$

(iii) If $|\mathbb{C}| = \emptyset$, then $\mathcal{T}_\sigma(\mathbb{C})$ is denoted by $\mathcal{T}_\sigma$, and $\mathcal{T}(\mathbb{C})$ is denoted by $\mathcal{T}$. The terms in $\mathcal{T}$ are called the pure (simply) typed λ-terms. $\square$

NOTATION. (i) For λ-terms we will use $M,N,L,P,Q,\ldots$

(ii) Outermost parentheses of λ-terms are usually omitted.

(iii) For application we use association to the left, so $MN_1N_2\ldots N_m$ denotes the λ-term $(\ldots((MN_1)N_2)\ldots)N_m$.

(iv) For multiple abstraction we use the following notation: $\lambda x_1x_2\ldots x_m.M$ denotes the λ-term $\lambda x_1.(\lambda x_2.(\ldots(\lambda x_m.M)\ldots))$. $\square$

1.1.7. DEFINITION. (i) For each $M \in \mathcal{T}(\mathbb{C})$ the set $FV(M)$ of all free variables of M is defined by induction on the structure of M as follows.

(1) $FV(x) = \{x\}$.

(2) $FV(\mathbf{C}) = \emptyset$.

(3) $FV(MN) = FV(M) \cup FV(N)$.

(4) $FV(\lambda x.M) = FV(M) - \{x\}$.

(ii) For each $M \in \mathcal{T}(\mathbb{C})$ the set $BV(M)$ of all bound variables of M is defined by induction on the structure of M as follows.

(1) $BV(x) = \emptyset$.

(2) $BV(\mathbf{C}) = \emptyset$.

(3) $BV(MN) = BV(M) \cup BV(N)$.

(4) $BV(\lambda x.M) = BV(M) \cup \{x\}$.

(iii) For each $M \in \mathcal{T}(\mathbb{C})$ the set $Var(M)$ of all variables of M is defined by

$$Var(M) = FV(M) \cup BV(M).$$

(iv) If $FV(M) = \emptyset$, then M is called closed. The set of all closed terms in $\mathcal{T}_\sigma(\mathbb{C})$ is denoted by $\mathcal{T}_\sigma(\mathbb{C})^0$, and the set of all closed terms in $\mathcal{T}(\mathbb{C})$ is denoted by $\mathcal{T}(\mathbb{C})^0$. $\square$

Two λ-terms that differ only in bound variables are called (syntactically) congruent. If M and N are congruent, we write $M \equiv N$. For example $\lambda x.x \equiv$

$\lambda y.y$ if x and y are variables of the same type. In the following we will define the congruence relation $\equiv$. The heart of the congruence relation is:

$$\lambda x.M \equiv \lambda y.M[x := y]$$

where $x, y \in Var_\sigma$, for some $\sigma \in \mathbb{T}$, with $y \notin Var(M)$, and where $M[x := y]$ denotes the result of substituting y in all free occurrences of x in M. But in order to define the substitution $M[x := N]$ we need to be able to 'change' bound variables. Therefore we will first define a restricted substitution $M(x := y)$ for variables $y \notin Var(M)$. Using this restricted substitution we can then define the congruence relation $\equiv$.

The λ-terms without a fixed variable can be defined inductively as follows.

1.1.8. DEFINITION. Let $y \in Var_{\mathbb{T}}$.

(i) For every $\sigma \in \mathbb{T}$ the set $\mathcal{T}_\sigma(\mathbb{C})^y$ of all type-σ λ-terms without y is defined inductively as follows.

(1) $x \in Var_\sigma \;\&\; x \neq y \Rightarrow x \in \mathcal{T}_\sigma(\mathbb{C})^y$.

(2) $\mathbb{C}_\sigma \subseteq \mathcal{T}_\sigma(\mathbb{C})^y$.

(3) $M \in \mathcal{T}_{\sigma\to\tau}(\mathbb{C})^y \;\&\; N \in \mathcal{T}_\sigma(\mathbb{C})^y \Rightarrow (MN) \in \mathcal{T}_\tau(\mathbb{C})^y$.

(4) $x \in Var_\sigma \;\&\; x \neq y \;\&\; M \in \mathcal{T}_\tau(\mathbb{C})^y \Rightarrow (\lambda x.M) \in \mathcal{T}_{\sigma\to\tau}(\mathbb{C})^y$.

(ii) The set of all λ-terms without y:

$$\mathcal{T}(\mathbb{C})^y = \bigcup_{\sigma\in\mathbb{T}} \mathcal{T}_\sigma(\mathbb{C})^y. \;\square$$

1.1.9. LEMMA. (i) $Var(x) = \{x\}$.

(ii) $Var(\mathbf{C}) = \emptyset$.

(iii) $Var(MN) = Var(M) \cup Var(N)$.

(iv) $Var(\lambda x.M) = Var(M) \cup \{x\}$.

PROOF. By 1.1.7. $\square$

1.1.10. LEMMA. $\mathcal{T}(\mathbb{C})^y = \{M \in \mathcal{T}(\mathbb{C}) | y \notin Var(M)\}$.

PROOF. By induction on the structure of M, using 1.1.9, it can be shown that

(i) $\forall M \in \mathcal{T}(\mathbb{C})^y[M \in \mathcal{T}(\mathbb{C}) \;\&\; y \notin Var(M)]$.

(ii) $\forall M \in \mathcal{T}(\mathbb{C})[y \notin Var(M) \Rightarrow M \in \mathcal{T}(\mathbb{C})^y]$. $\square$

1.1.11. DEFINITION. Let $\sigma \in \mathbb{T}$, $x, y \in Var_\sigma$. For every $M \in \mathcal{T}(\mathbb{C})^y$ define $M(x := y) \in \mathcal{T}(\mathbb{C})$ by induction on the structure of M as follows.

(i) $x(x := y) = y$.

(ii) $z(x := y) = z$ if $z \neq x$.

(iii) $\mathbf{C}(x := y) = \mathbf{C}$.
(iv) $(MN)(x := y) = (M(x := y)N(x := y))$.
(v) $(\lambda x.M)(x := y) = (\lambda x.M)$.
(vi) $(\lambda z.M)(x := y) = (\lambda z.M(x := y))$ if $z \neq x$. □

1.1.12. DEFINITION. (i) The congruence relation $\equiv$ is defined inductively as follows.

(1) $\lambda x.M \equiv \lambda y.M(x := y)$ if $y \notin Var(M)$.

(2) $M \equiv M$.

(3) $M \equiv N \Rightarrow N \equiv M$.

(4) $M \equiv N \;\&\; N \equiv L \Rightarrow M \equiv L$.

(5) $M \equiv N \Rightarrow ML \equiv NL$.

(6) $M \equiv N \Rightarrow LM \equiv LN$.

(7) $M \equiv N \Rightarrow \lambda x.M \equiv \lambda x.N$.

(ii) If $M \equiv N$, then M and N are called congruent. □

1.1.13. LEMMA. *For every $M \in \mathcal{T}(\mathbb{C})$ and every finite set V of variables, there exists $N \in \mathcal{T}(\mathbb{C})$ such that*

$$M \equiv N \;\&\; BV(N) \cap V = \emptyset.$$

PROOF. By induction on the structure of M.

(i) If M is a variable or a constant, take $N = M$.

(ii) Let $M = M_1M_2$, and let $V \subseteq Var_{\mathbb{T}}$ be finite. By induction there exist terms N_1 and N_2 with $M_1 \equiv N_1, M_2 \equiv N_2, BV(N_1) \cap V = \emptyset$ and $BV(N_2) \cap V = \emptyset$. Take $N = N_1N_2$. Then $M \equiv N$ and $BV(N) \cap V = \emptyset$.

(iii) Let $M = \lambda x.M$, with $x \in Var_\sigma$, and let $V \subseteq Var_{\mathbb{T}}$ be finite. Then $W = V \cup \{x\}$ is also finite, and by induction there exists N_1 with $M_1 \equiv N_1$ and $BV(N_1) \cap W = \emptyset$. Then $x \notin BV(N_1)$ and $BV(N_1) \cap V = \emptyset$. Now let $y \in Var_\sigma$ be such that $y \notin V$. Take $N = \lambda y.N_1(x := y)$. Then $M \equiv N$ and $BV(N) \cap V = \emptyset$, because $BV(N) = BV(N_1) \cup \{y\}$. □

Most of the time we will work with λ-terms modulo congruence, because the exact choice for the bound variables is unimportant; it only matters to know which congruence class of terms one is working with. The Variable Convention states that in every practical situation the bound variables are chosen to be different from the free variables. The previous lemma implies that it is possible to obey this convention, and we will always tacitly do so.

1.1.14. DEFINITION. Let $\sigma \in \mathbb{T}, x \in Var_\sigma, N \in \mathcal{T}_\sigma(\mathbb{C})$. For every $M \in \mathcal{T}(\mathbb{C})$ the substitution $M[x := N]$ is defined by induction on the structure of M as follows.

(i) $x[x := N] \equiv N$.
(ii) $y[x := N] \equiv y$ if $y \not\equiv x$.
(iii) $(M_1 M_2)[x := N] \equiv M_1[x := N]M_2[x := N]$.
(iv) $(\lambda y.M)[x := N] \equiv \lambda y.M[x := N]$. □

1.2. Systems and reduction.

A simply typed λ-calculus or system consists of a set $\mathcal{T}(\mathbb{C})$ of simply typed λ-terms together with a set of reduction rules. According to these reduction rules subterms of λ-terms can be replaced by other λ-terms. This is called reduction.

A reduction rule on $\mathcal{T}(\mathbb{C})$ is a pair (M, N) of λ-terms in $\mathcal{T}(\mathbb{C})$ of the same type with $FV(N) \subseteq FV(M)$. A reduction rule (M, N) is also written as $M \to N$. A set of reduction rules is called a reductor.

1.2.1. Definition. A system is a pair $\Sigma = (\mathcal{T}(\mathbb{C}), \mathbf{R})$ where $\mathbf{R}$ is a reductor on $\mathcal{T}(\mathbb{C})$. □

If $\Sigma = (\mathcal{T}(\mathbb{C}), \mathbf{R})$ is a system, then $\mathcal{T}(\mathbb{C})$ is also denoted by $\mathcal{T}(\Sigma)$, and the elements of $\mathcal{T}(\Sigma)$ are called Σ-terms. We write further $\mathcal{T}_\sigma(\Sigma) = \mathcal{T}_\sigma(\mathbb{C})$, $\mathcal{T}(\Sigma)^0 = \mathcal{T}(\mathbb{C})^0$ and $\mathcal{T}_\sigma(\Sigma)^0 = \mathcal{T}_\sigma(\mathbb{C})^0$.

1.2.2. Definition. Let $\mathbf{R}$ be a reductor on $\mathcal{T}(\mathbb{C})$.

(i) The reduction rules in $\mathbf{R}$ are called R-rules. If $M \to N$ is an R-rule, then we write $\mathbf{R} : M \to N$.

(ii) If $\mathbf{R} : M \to N$, then M is an R-redex, and N is a contractum of M.

(iii) The reductor $\mathbf{R}$ is deterministic if

$$\mathbf{R} : M \to N_1 \;\&\; \mathbf{R} : M \to N_2 \Rightarrow N_1 \equiv N_2. \;\square$$

If $\mathbf{R}_1, \mathbf{R}_2, \ldots, \mathbf{R}_n$ are reductors on $\mathcal{T}(\mathbb{C})$, then their union $\mathbf{R}_1 \cup \mathbf{R}_2 \cup \ldots \mathbf{R}_n$ is also a reductor, and is denoted by $\mathbf{R}_1\mathbf{R}_2 \ldots \mathbf{R}_n$. So $\mathbf{R}_1\mathbf{R}_2 = \mathbf{R}_1 \cup \mathbf{R}_2$, $\mathbf{R}_1\mathbf{R}_2\mathbf{R}_3 = \mathbf{R}_1 \cup \mathbf{R}_2 \cup \mathbf{R}_3$, etcetera.

The reductor β on $\mathcal{T}(\mathbb{C})$ consists of all reduction rules of the form

$$(\lambda x.M)N \to M[x := N].$$

Note that $FV(M[x := N]) \subseteq FV((\lambda x.M)N)$, so β is indeed a reductor. Strictly speaking we have a different reductor β for every set $\mathcal{T}(\mathbb{C})$, but every one of these is denoted by β.

The reductor η on $\mathcal{T}(\mathbb{C})$ consists of all reduction rules of the form

$$\lambda x.Mx \to M$$

with $x \notin FV(M)$.

On every set $\mathcal{T}(\mathbb{C})$ we now have the reductors β, η and $\beta\eta = \beta \cup \eta$. For every $\mathbb{C}$ we have two systems:

- $\lambda^{\rightarrow}(\mathbb{C}) = (\mathcal{T}(\mathbb{C}), \beta)$
- $\lambda^{\rightarrow}\eta(\mathbb{C}) = (\mathcal{T}(\mathbb{C}), \beta\eta)$.

Recall that $\mathcal{T} = \mathcal{T}(\mathbb{C})$ where $|\mathbb{C}| = \emptyset$, so in particular we have the systems:

- $\lambda^{\rightarrow} = (\mathcal{T}, \beta)$
- $\lambda^{\rightarrow}\eta = (\mathcal{T}, \beta\eta)$.

The system $\lambda^{\rightarrow}$ is called the pure simply typed λ-calculus, and the system $\lambda^{\rightarrow}\eta$ is called the pure extensional simply typed λ-calculus.

1.2.3. Definition. Let $\Sigma = (\mathcal{T}(\mathbb{C}), \mathbf{R})$ be a system. Then Σ is called a β-system if $\beta \subseteq \mathbf{R}$, and Σ is called a $\beta\eta$-system if $\beta\eta \subseteq \mathbf{R}$. $\Box$

Let $\mathbf{R}$ be a reductor on $\mathcal{T}(\mathbb{C})$. An R-reduction step consists of replacing an R-redex occurrence in a λ-term by a contractum of that redex. If a redex is being replaced by a contractum then that redex is said to be contracted. In the following we define the one-step R-reduction relation $\rightarrow_R$ on $\mathcal{T}(\mathbb{C})$. But first we define subterms and subterm occurrences. For this we need contexts. A context is a λ-term with a hole in it. In such a hole one can put a λ-term.

1.2.4. Definition. (i) For every $\upsilon \in \mathbb{T}$ take a new constant $\Box^{\upsilon}$, called the hole of type υ.

(ii) Let $\upsilon \in \mathbb{T}$. An υ-context will be a context containing the hole of type υ. For every $\sigma \in \mathbb{T}$ the set $Context^{\upsilon}_{\sigma}(\mathbb{C})$ of υ-contexts of type σ is defined inductively as follows.

(1) $\Box^{\upsilon} \in Context^{\upsilon}_{\upsilon}(\mathbb{C})$.

(2) $C \in Context^{\upsilon}_{\sigma\rightarrow\tau}(\mathbb{C})\ \&\ M \in \mathcal{T}_{\sigma}(\mathbb{C}) \Rightarrow (CM) \in Context^{\upsilon}_{\tau}(\mathbb{C})$.

(3) $M \in \mathcal{T}_{\sigma\rightarrow\tau}(\mathbb{C})\ \&\ C \in Context^{\upsilon}_{\sigma}(\mathbb{C}) \Rightarrow (MC) \in Context^{\upsilon}_{\tau}(\mathbb{C})$.

(4) $x \in Var_{\sigma}\ \&\ C \in Context^{\upsilon}_{\tau}(\mathbb{C}) \Rightarrow (\lambda x.C) \in Context^{\upsilon}_{\sigma\rightarrow\tau}(\mathbb{C})$.

(iii) The set of all υ-contexts:

$$Context^{\upsilon}(\mathbb{C}) = \bigcup_{\sigma \in \mathbb{T}} Context^{\upsilon}_{\sigma}(\mathbb{C}).$$

(iv) The set of all contexts with constants from $\mathbb{C}$ is denoted by $Context(\mathbb{C})$.

(v) Let $\upsilon \in \mathbb{T}$, $N \in \mathcal{T}_{\upsilon}(\mathbb{C})$. For every υ-context C define $C[N] \in \mathcal{T}(\mathbb{C})$ by induction on the structure of C as follows.

(1) $\Box^{v}[N] = N$.
(2) $(CM)[N] = (C[N]M)$.
(3) $(MC)[N] = (MC[N])$.
(4) $(\lambda x.C)[N] = (\lambda x.C[N])$. $\Box$

NOTATION. For contexts we will use $C, D, E, C_1, C_2, \ldots$ $\Box$

1.2.5. DEFINITION. Let $M \in \mathcal{T}(\mathbb{C})$.

(i) Let $N \in \mathcal{T}(\mathbb{C})$. Then N is a subterm of M (notation $N \subseteq M$) if there exists a context C such that $C[N] = M$.

(ii) Let $N \subseteq M$. An occurrence of N in M is a pair (C, N) such that C is a context with $C[N] = M$.

(iii) Let $x \in Var_{\mathbb{T}}$. Then $x \in M \iff x \subseteq M$.

(iv) Let $\mathbf{C} \in |\mathbb{C}|$. Then $\mathbf{C} \in M \iff \mathbf{C} \subseteq M$. $\Box$

In the hole of a σ-context one can also put a context of type σ.

1.2.6. DEFINITION. Let $\sigma, v \in \mathbb{T}$, $D \in Context_{\sigma}^{v}(\mathbb{C})$. For every σ-context C the v-context $C[D]$ is defined by induction on the structure of C as follows.

(i) $\Box^{\sigma}[D] = D$.
(ii) $(CM)[D] = (C[D]M)$.
(iii) $(MC)[D] = (MC[D])$.
(iv) $(\lambda x.C)[D] = (\lambda x.C[D])$. $\Box$

1.2.7. LEMMA. (i) $C[D][M] = C[D[M]]$.
(ii) $C[D][E] = C[D[E]]$.
(iii) $C[M] = C[N] \Rightarrow M = N$.
(iv) $C[D] = C[E] \Rightarrow D = E$.

PROOF. Simple, by induction on the structure of C. $\Box$

1.2.8. DEFINITION. Let $M \in \mathcal{T}(\mathbb{C})$ and let (C_1, N_1) and (C_2, N_2) be subterm occurrences in M.

(i) (C_1, N_1) is contained in (C_2, N_2) (notation $(C_1, N_1) \subseteq (C_2, N_2)$) if there exists a context D such that $C_1 = C_2[D]$.

(ii) (C_1, N_1) and (C_2, N_2) are called disjoint if neither is contained in the other. $\Box$

1.2.9. LEMMA. *Let (C_1, N_1) and (C_2, N_2) be subterm occurrences in M such that $(C_1, N_1) \subseteq (C_2, N_2)$. Then there exists a unique context D such that $C_1 = C_2[D]$. For this D we have that (D, N_1) is a subterm occurrence in N_2.*

PROOF. By lemma 1.2.7. $\Box$

1.2.10. DEFINITION. Let $\mathbf{R} \subseteq \mathcal{T}(\mathbb{C})^2$ be a binary relation on λ-terms.
(i) The relation $\mathbf{R}$ is type-preserving if

$$(M,N) \in \mathbf{R} \;\&\; M \in \mathcal{T}_\sigma(\mathbb{C}) \Rightarrow N \in \mathcal{T}_\sigma(\mathbb{C}).$$

(ii) The relation $\mathbf{R}$ is compatible if $\mathbf{R}$ is type-preserving and
(1) $(M,N) \in \mathbf{R} \Rightarrow (ML, NL) \in \mathbf{R}$.
(2) $(M,N) \in \mathbf{R} \Rightarrow (LM, LN) \in \mathbf{R}$.
(3) $(M,N) \in \mathbf{R} \Rightarrow (\lambda x.M, \lambda x.N) \in \mathbf{R}$.
(iii) The relation $\mathbf{R}$ is syntactic if $\mathbf{R}$ is type-preserving and

$$(M_1, N_1) \in \mathbf{R} \;\&\; M_1 \equiv M_2 \;\&\; N_1 \equiv N_2 \Rightarrow (M_2, N_2) \in \mathbf{R}. \;\Box$$

1.2.11. DEFINITION. Let $\mathbf{R}$ be type-preserving.
(i) The compatible closure of $\mathbf{R}$ is:

$$com(\mathbf{R}) = \{(C[M], C[N]) | C \in Context(\mathbb{C}) \;\&\; (M,N) \in \mathbf{R}\}.$$

(ii) The syntactic closure of $\mathbf{R}$ is:

$$syn(\mathbf{R}) = \{(M,N) | \exists (M_0, N_0) \in \mathbf{R}[M \equiv M_0 \;\&\; N \equiv N_0]\}. \;\Box$$

1.2.12. LEMMA. *Let* $\mathbf{R}$ *be type-preserving.*
(i) $com(\mathbf{R})$ *is the least compatible relation containing* $\mathbf{R}$.
(ii) $syn(\mathbf{R})$ *is the least syntactic relation containing* $\mathbf{R}$.
(iii) *If* $\mathbf{R}$ *is compatible, then* $syn(\mathbf{R})$ *is also compatible.*
(iv) $syn(com(\mathbf{R}))$ *is the least syntactic and compatible relation containing* $\mathbf{R}$.

PROOF. Simple. $\Box$

The one-step R-reduction relation $\rightarrow_R$ on $\mathcal{T}(\mathbb{C})$ is obtained by taking the syntactic and compatible closure of the reductor $\mathbf{R}$: for all $M, N \in \mathcal{T}(\mathbb{C})$

$$M \rightarrow_R N \iff (M,N) \in syn(com(R)).$$

If $M \rightarrow_R N$, then M is said to R-reduce to N in one step. The binary relation $\rightarrow_R$ is type-preserving, compatible and syntactic. By the definition of the syntactic and compatible closures we obtain the following lemma.

1.2.13. LEMMA. $M \rightarrow_R N$ *if and only if there exists a context* C *and an* R*-rule* $P \rightarrow Q$ *such that* $M \equiv C[P]$ *and* $N \equiv C[Q]$. $\Box$

1.2.14. DEFINITION. (i) An R-redex occurrence in M is an occurrence $O = (C, P)$ in M such that P is an R-redex.

(ii) Let $O = (C, P)$ be an R-redex occurrence in M. Then we write

$$M \to_R^O N$$

if there exists Q with $\mathbf{R} : P \to Q$ and $N \equiv C[Q]$. $\Box$

Next we define R-reduction $\twoheadrightarrow_R$ and R-conversion $=_R$ on $\mathcal{T}(\mathbb{C})$.

1.2.15. Definition. (i) R-Reduction is the relation $\twoheadrightarrow_R$ on $\mathcal{T}(\mathbb{C})$ defined inductively by

(1) $M \to_R N \Rightarrow M \twoheadrightarrow_R N$

(2) $M \equiv N \Rightarrow M \twoheadrightarrow_R N$

(3) $M \twoheadrightarrow_R N \;\&\; N \twoheadrightarrow_R L \Rightarrow M \twoheadrightarrow_R L$.

(ii) R-Conversion or R-equality is the relation $=_R$ on $\mathcal{T}(\mathbb{C})$ defined inductively by

(1) $M \twoheadrightarrow_R N \Rightarrow M =_R N$

(2) $M =_R N \Rightarrow N =_R M$

(3) $M =_R N \;\&\; N =_R L \Rightarrow M =_R L$. $\Box$

1.2.16. Definition. (i) A term M is called an R-normal form if there does not exist a term N with $M \to_R N$.

(ii) A term M has an R-normal form N if N is an R-normal form and $M =_R N$. $\Box$

Next we define a few special λ-terms.

1.2.17. Definition. (i) $\mathbf{I}_\sigma \equiv \lambda x.x$ with $x \in Var_\sigma$.

(ii) $\mathbf{K}_{\sigma,\tau} \equiv \lambda xy.x$ with $x \in Var_\sigma$ and $y \in Var_\tau$.

(iii) $\mathbf{S}_{\sigma,\tau,\upsilon} \equiv \lambda xyz.xz(yz)$ with $x \in Var_{\sigma\to\tau\to\upsilon}$, $y \in Var_{\sigma\to\tau}$ and $z \in Var_\sigma$. $\Box$

1.2.18. Lemma. (i) $\mathbf{I}_\sigma M \twoheadrightarrow_\beta M$.

(ii) $\mathbf{K}_{\sigma,\tau} MN \twoheadrightarrow_\beta M$.

(iii) $\mathbf{S}_{\sigma,\tau,\upsilon} MNL \twoheadrightarrow_\beta ML(NL)$. $\Box$

1.2.19. Definition. Let $\mathbf{R}$ be a reductor on $\mathcal{T}(\mathbb{C})$. Then $\mathbf{R}$ has the Church-Rosser property (or the $\mathbf{CR}$-property) if for all $M, M_1, M_2 \in \mathcal{T}(\mathbb{C})$ with $M \twoheadrightarrow_R M_1$ and $M \twoheadrightarrow_R M_2$ there exists $N \in \mathcal{T}(\mathbb{C})$ such that $M_1 \twoheadrightarrow_R N$ and $M_2 \twoheadrightarrow_R N$.

$$\begin{tikzcd}
M \arrow[r, two heads, "R"] \arrow[d, two heads, "R"'] & M_1 \arrow[d, two heads, "R"] \\
M_2 \arrow[r, two heads, "R"'] & N
\end{tikzcd}$$

If $\mathbf{R}$ has the $\mathbf{CR}$-property, then we write $\mathbf{CR}(\mathbf{R})$. If $\Sigma = (\mathcal{T}(\mathbb{C}), \mathbf{R})$, then Σ has the $\mathbf{CR}$-property if $\mathbf{R}$ has the $\mathbf{CR}$-property and we write $\Sigma \models \mathbf{CR}$. $\Box$

1.2.20. THEOREM. $\lambda^{\rightarrow}(\mathbb{C}) \models \mathbf{CR}$. $\Box$

The proof of this theorem is similar to the proof of $\mathbf{CR}(\beta)$ in the type-free λ-calculus; see for example [Barendregt 1984].

The next theorem states a few well-known consequences of the Church-Rosser property.

1.2.21. THEOREM. *Let $\Sigma = (\mathcal{T}(\mathbb{C}), \mathbf{R})$ be a system with $\Sigma \models \mathbf{CR}$.*

(i) *If $M_1 =_R M_2$, then there exists $M \in \mathcal{T}(\mathbb{C})$ such that $M_1 \twoheadrightarrow_R M$ and $M_2 \twoheadrightarrow_R M$.*

(ii) *If M has R-normal form N, then $M \twoheadrightarrow_R N$.*

(iii) *If M_1 and M_2 are R-normal forms with $M_1 =_R M_2$, then $M_1 \equiv M_2$.* $\Box$

Sometimes it is useful to view a system as an equational theory.

1.2.22. DEFINITION. An equation on $\mathcal{T}(\mathbb{C})$ is any expression of the form $(M = N)$ with $M, N \in \mathcal{T}_\sigma(\mathbb{C})$ for some $\sigma \in \mathbb{T}$. $\Box$

Let Γ be a set of equations on $\mathcal{T}(\mathbb{C})$. If we look at Γ as a set of axioms, then $Th(\Gamma)$ is the set of all theorems that can be derived from Γ.

1.2.23. DEFINITION. (i) The set $Th(\Gamma)$ of all theorems of Γ is defined inductively as follows.

(1) $\Gamma \subseteq Th(\Gamma)$

(2) $M \equiv N \Rightarrow (M = N) \in Th(\Gamma)$

(3) $(M = N) \in Th(\Gamma) \Rightarrow (N = M) \in Th(\Gamma)$

(4) $(M = N), (N = L) \in Th(\Gamma) \Rightarrow (M = L) \in Th(\Gamma)$

(5) $(M = N) \in Th(\Gamma) \Rightarrow (ML = NL) \in Th(\Gamma)$

(6) $(M = N) \in Th(\Gamma) \Rightarrow (LM = LN) \in Th(\Gamma)$

(7) $(M = N) \in Th(\Gamma) \Rightarrow (\lambda x.M = \lambda x.N) \in Th(\Gamma)$.

(ii) For all $\sigma \in \mathbb{T}$ and $M, N \in \mathcal{T}_\sigma(\mathbb{C})$ define

$$\Gamma \vdash M = N \iff (M = N) \in Th(\Gamma). \ \Box$$

A theory is a set of equations that is closed under derivability.

1.2.24. DEFINITION. Let Γ and Γ_1 be sets of equations.

(i) Γ_1 is a theory if $Th(\Gamma_1) \subseteq \Gamma_1$.

(ii) Γ_1 is a Γ-theory if Γ_1 is a theory with $\Gamma \subseteq \Gamma_1$. $\square$

1.2.25. Lemma. *A set of equations Γ is a theory if and only if the following hold.*

(i) $M \equiv N \Rightarrow (M = N) \in \Gamma$
(ii) $(M = N) \in \Gamma \Rightarrow (N = M) \in \Gamma$
(iii) $(M = N), (N = L) \in \Gamma \Rightarrow (M = L) \in \Gamma$
(iv) $(M = N) \in \Gamma \Rightarrow (ML = NL) \in \Gamma$
(v) $(M = N) \in \Gamma \Rightarrow (LM = LN) \in \Gamma$
(vi) $(M = N) \in \Gamma \Rightarrow (\lambda x.M = \lambda x.N) \in \Gamma$. $\square$

1.2.26. Lemma. *$Th(\Gamma)$ is the least Γ-theory.* $\square$

1.2.27. Definition. Let $\Sigma = (\mathcal{T}(\mathbb{C}), \mathbf{R})$ be a system.

(i) The set of all Σ-axioms or axioms of Σ is defined by

$$Ax(\Sigma) = \{(M = N) \mid \mathbf{R} : M \to N\}.$$

(ii) The set of all Σ-theorems or theorems of Σ is defined by

$$Th(\Sigma) = Th(Ax(\Sigma)).$$

(iii) For all $M, N \in \mathcal{T}(\Sigma)$

$$\Sigma \vdash M = N \iff (M = N) \in Th(\Sigma).$$

If $\Sigma \vdash M = N$, then we say that Σ proves $M = N$, or that in Σ we have $M = N$.

(iv) A Σ-theory is a theory Γ with $Ax(\Sigma) \subseteq \Gamma$. $\square$

1.2.28. Lemma. *$Th(\Sigma)$ is the least Σ-theory.* $\square$

1.2.29. Theorem. *Let $\Sigma = (\mathcal{T}(\mathbb{C}), \mathbf{R})$ be a system. Then*

$$\Sigma \vdash M = N \iff M =_R N.$$

Proof. (i) Define $\Gamma = \{(M = N) \mid M =_R N\}$. Then Γ is a Σ-theory and so $Th(\Sigma) \subseteq \Gamma$.

(ii) Define $R_1 = \{(M, N) \mid \Sigma \vdash M = N\}$. Then R_1 is a compatible and syntactic relation which contains R, so $\to_R \subseteq R_1$. But R_1 is also an equivalence relation which contains $\equiv$ and so $\twoheadrightarrow_R \subseteq R_1$ and $=_R \subseteq R_1$. $\square$

1.3. Delta-reduction.

Delta-reduction can be used to internalize functions on $\mathcal{T}(\mathbb{C})$. If φ is a function on $\mathcal{T}(\mathbb{C})$, say

$$\varphi : \mathcal{T}_{\sigma_1}(\mathbb{C}) \times \ldots \times \mathcal{T}_{\sigma_m}(\mathbb{C}) \rightarrow \mathcal{T}_\tau(\mathbb{C})$$

then φ can be internalized by taking a constant $\mathbf{C} \in \mathbb{C}_{\sigma_1 \rightarrow \cdots \rightarrow \sigma_m \rightarrow \tau}$ and adding reduction rules

$$\mathbf{C} M_1 \ldots M_m \rightarrow \varphi(M_1, \ldots, M_m).$$

This kind of reduction is called δ-reduction. In fact, our definition of δ-reduction will be a bit more general than this, and will also include non-deterministic reduction.

1.3.1. Definition. A δ-reductor on $\mathcal{T}(\mathbb{C})$ is a reductor δ on $\mathcal{T}(\mathbb{C})$ such that every δ-redex is of the form $\mathbf{C} M_1 \ldots M_m$ with $\mathbf{C} \in |\mathbb{C}|$ and there exists a function $arity : |\mathbb{C}| \rightarrow \omega$ such that

(i) if $\mathbf{C} M_1 \ldots M_m$ is a δ-redex, then $arity(\mathbf{C}) = m$

(ii) for all $\sigma \in \mathbb{T}$ and $\mathbf{C} \in \mathbb{C}_\sigma$ we have that $arity(\mathbf{C}) \leq fac(\sigma)$. $\square$

1.3.2. Definition. (i) A $\beta\delta$-system is a system $\Sigma = (\mathcal{T}(\mathbb{C}), \beta\delta)$ where δ is a δ-reductor on $\mathcal{T}(\mathbb{C})$.

(ii) A $\beta\eta\delta$-system is a system $\Sigma = (\mathcal{T}(\mathbb{C}), \beta\eta\delta)$ where δ is a δ-reductor on $\mathcal{T}(\mathbb{C})$. $\square$

A δ-axiom is an equation of the form $(\mathbf{C} M_1 \ldots M_m = N)$. If Σ is a $\beta\delta$-system or a $\beta\eta\delta$-system, then $Ax_\delta(\Sigma)$ is the set of all δ-axioms of Σ.

A well-known example of a $\beta\delta$-system is the system $\lambda\mathsf{T}$ which is a λ-calculus variant of Gödel's system T of primitive recursive functionals.

1.3.3. Definition. The system $\lambda\mathsf{T}$ is defined as follows.

(i) The constants of $\lambda\mathsf{T}$, their types and their arities are given by the following table.

constant	type	arity
$\mathbf{c}_m$ $(m \in \omega)$	0	0
$\mathbf{S}^+$	$0 \rightarrow 0$	1
$\mathbf{R}_\sigma$ $(\sigma \in \mathbb{T})$	$\sigma \rightarrow (\sigma \rightarrow 0 \rightarrow \sigma) \rightarrow 0 \rightarrow \sigma$	3

This table already gives the set $\mathcal{T}(\lambda\mathsf{T})$ of all $\lambda\mathsf{T}$-terms.

(ii) Let δ be the reductor on $\mathcal{T}(\lambda\mathsf{T})$ consisting of the following reduction rules:

(1) $\mathbf{S}^+ \mathbf{c}_m \rightarrow \mathbf{c}_{m+1}$.

(2) $\mathbf{R}_\sigma M N \mathbf{c}_0 \rightarrow M$.

(3) $\mathbf{R}_\sigma M N \mathbf{c}_{m+1} \rightarrow N(\mathbf{R}_\sigma M N \mathbf{c}_m)\mathbf{c}_m$.

(iii) $\lambda\mathsf{T} = (\mathcal{T}(\lambda\mathsf{T}), \beta\delta)$. $\square$

Another well-known example of a $\beta\delta$-system is $\lambda\mathsf{B}$, which consists of $\lambda\mathsf{T}$ extended with bar recursion; see for example [Spector 1962] or [Bezem 1986].

Using δ-reduction we can also construct systems with fixed point combinators, thus incorporating full recursion into simply typed λ-calculus.

A fixed point combinator (for type σ) is a term Y_σ of type $(\sigma\to\sigma)\to\sigma$ such that for every term F of type $\sigma\to\sigma$ we have

$$\mathsf{Y}_\sigma F = F(\mathsf{Y}_\sigma F)$$

so that $\mathsf{Y}_\sigma F$ is a fixed point of F.

In the following we will construct $\beta\delta$-systems with fixed point combinators. For every $\sigma\in\mathbb{T}$ take a constant $\mathbf{Y}_\sigma$ of type $(\sigma\to\sigma)\to\sigma$. These constants will perform δ-reduction. This kind of δ-reduction will be called γ-reduction. The first thing that comes to mind, if one wishes to define γ-reduction, is setting:

$$\gamma : \mathbf{Y}_\sigma F \to F(\mathbf{Y}_\sigma F)$$

for every F of type $\sigma\to\sigma$. But this choice for the definition of γ has a disadvantage. Although we have $\mathbf{CR}(\beta\gamma)$ (by Klop's theorem on the confluency of regular CRS's; see [Klop 1980]), we do not have $\mathbf{CR}(\beta\eta\gamma)$, by the following counterexample.

$$\begin{tikzcd}
\lambda f.\mathbf{Y}_\sigma f \arrow[r, "\gamma"] \arrow[d, "\eta"'] & \lambda f.f(\mathbf{Y}_\sigma f) \\
\mathbf{Y}_\sigma &
\end{tikzcd}$$

The only way to reduce $\lambda f.f(\mathbf{Y}_\sigma f)$ is:

$$\begin{aligned}
\lambda f.f(\mathbf{Y}_\sigma f) &\to_\gamma \lambda f.f(f(\mathbf{Y}_\sigma f)) \\
&\to_\gamma \lambda f.f(f(f(\mathbf{Y}_\sigma f))) \\
&\to_\gamma \cdots
\end{aligned}$$

So the terms $\mathbf{Y}_\sigma$ and $\lambda f.f(\mathbf{Y}_\sigma f)$ do not have a common $\beta\eta\gamma$-reduct, and therefore $\neg\mathbf{CR}(\beta\eta\gamma)$. For this reason we define γ a bit differently.

1.3.4. DEFINITION. The systems $\lambda\mathsf{Y}$ and $\lambda\mathsf{Y}\eta$ are defined as follows.

(i) The constants of $\lambda\mathsf{Y}$, their types and their arities are given by the following table.

constant	type	arity
$\mathbf{Y}_\sigma$ $(\sigma \in \mathbb{T})$	$(\sigma \to \sigma) \to \sigma$	0

This table gives a set $\mathcal{T}(\lambda\mathsf{Y})$ of $\lambda\mathsf{Y}$-terms.

(ii) Let γ be the reductor on $\mathcal{T}(\lambda\mathsf{Y})$ consisting of the following reduction rules:

$$\mathbf{Y}_\sigma \to \lambda f.f(\mathbf{Y}_\sigma f)$$

with $f \in Var_{\sigma\to\sigma}$.

(iii) $\lambda\mathsf{Y} = (\mathcal{T}(\lambda\mathsf{Y}), \beta\gamma)$.

(iv) $\lambda\mathsf{Y}\eta = (\mathcal{T}(\lambda\mathsf{Y}), \beta\gamma\eta)$. $\Box$

1.3.5. Lemma. *For every* $F \in \mathcal{T}_{\sigma\to\sigma}(\lambda\mathsf{Y})$

$$\mathbf{Y}_\sigma F \twoheadrightarrow_{\beta\gamma} F(\mathbf{Y}_\sigma F).$$

Proof. Let $F \in \mathcal{T}_{\sigma\to\sigma}(\lambda\mathsf{Y})$. Then

$$\begin{array}{rcl} \mathbf{Y}_\sigma F & \to_\gamma & (\lambda f.f(\mathbf{Y}_\sigma f))F \\ & \to_\beta & F(\mathbf{Y}_\sigma F). \ \Box \end{array}$$

A structure $\mathcal{A}$ consists of a set A of objects together with a number of relations and functions on A. Using δ-reduction we can add a structure $\mathcal{A}$ to $\lambda\mathsf{Y}$, obtaining the system $\lambda\mathsf{Y}(\mathcal{A})$. This kind of δ-reduction will be called α-reduction.

1.3.6. Definition. (i) A similarity type is a pair

$$\$ = ((p_1, p_2, \ldots, p_r), (q_1, q_2, \ldots, q_s))$$

with $r, s \in \omega$ and $p_1, \ldots, p_r, q_1, \ldots q_s \in \omega$.

(ii) Let $\$ = ((p_1, \ldots, p_r), (q_1, \ldots q_s))$ be a similarity type. A structure with similarity type $\$$ is a structure

$$\mathcal{A} = (A, R_1, \ldots, R_r, f_1, \ldots, f_s)$$

with

(1) A is a non-empty set.

(2) $R_i \subseteq A^{p_i}$ $(i = 1, \ldots, r)$.

(3) $f_j : A^{q_j} \to A$ $(j = 1, \ldots, s)$.

The set A is called the domain of $\mathcal{A}$. $\Box$

1.3.7. EXAMPLES. (i) The structure $(\omega, =, S^+)$ of the natural numbers together with the equality relation $=$ on ω and the successor function $S^+ : \omega \to \omega$ is a structure with similarity type $((2),(1))$.

(ii) $(\omega, \{0\}, S^+)$ is a structure with similarity type $((1),(1))$.

(iii) The ordered ring of the integers $(\mathbb{Z}, \leq, +, \cdot)$ is a structure with similarity type $((2),(2,2))$. $\square$

In order to construct the systems $\lambda\mathsf{Y}(\mathcal{A})$ we have to define a few special types and terms.

1.3.8. DEFINITION. (i) Define $t : \omega \to \mathbb{T}$ by $t(0) = 0$ and $t(n+1) = 0 \to t(n)$.

(ii) The booleans of type $\sigma \to \sigma \to \sigma$ are defined by

(1) $\mathbf{true}_\sigma \equiv \lambda xy.x$

(2) $\mathbf{false}_\sigma \equiv \lambda xy.y$

with $x, y \in Var_\sigma$.

(iii) For B of type $\sigma \to \sigma \to \sigma$ and for P and Q of type σ define

$$\mathbf{if}\ B\ \mathbf{then}\ P\ \mathbf{else}\ Q \equiv BPQ. \ \square$$

Let $\mathcal{A} = (A, R_1, \ldots, R_r, f_1 \ldots, f_s)$ be a structure with similarity type $\$ = ((p_1, \ldots, p_r), (q_1, \ldots, q_s))$. In the system $\lambda\mathsf{Y}(\mathcal{A})$ every element $a \in A$ will be represented by a constant $\mathbf{c}_a$ of type 0. The relations and functions of $\mathcal{A}$ will be represented as follows.

Let $R \subseteq A^p$ with $p \in \omega$. To represent R we take a constant $\mathbf{R}$ of type $t(p+2)$ and we add the reduction rules

- $\mathbf{R}\mathbf{c}_{a_1} \ldots \mathbf{c}_{a_p} \to \mathbf{true}_0$ for $(a_1, \ldots, a_p) \in R$.
- $\mathbf{R}\mathbf{c}_{a_1} \ldots \mathbf{c}_{a_p} \to \mathbf{false}_0$ for $(a_1, \ldots, a_p) \notin R$.

Let $f : A^q \to A$ with $q \in \omega$. To represent f we take a constant $\mathbf{F}$ of type $t(q)$ and we add the reduction rules

$$\mathbf{F}\mathbf{c}_{a_1} \ldots \mathbf{c}_{a_q} \to \mathbf{c}_{f(a_1,\ldots,a_q)}$$

for all $(a_1, \ldots, a_q) \in A$.

1.3.9. DEFINITION. The systems $\lambda\mathsf{Y}(\mathcal{A})$ and $\lambda\mathsf{Y}\eta(\mathcal{A})$ are defined as follows.

(i) The constants of $\lambda\mathsf{Y}(\mathcal{A})$, their types and their arities are given by the following table.

constant	type	arity
$\mathbf{c}_a$ $(a \in A)$	0	0
$\mathbf{R}_i$ $(i = 1, \ldots, r)$	$t(p_i + 2)$	p_i
$\mathbf{F}_j$ $(j = 1, \ldots, s)$	$t(q_j)$	q_j
$\mathbf{Y}_\sigma$ $(\sigma \in \mathbb{T})$	$(\sigma \to \sigma) \to \sigma$	0

By this table we have the set $\mathcal{T}(\lambda\mathsf{Y}(\mathcal{A}))$ of all $\lambda\mathsf{Y}(\mathcal{A})$-terms.

(ii) Let α be the reductor on $\mathcal{T}(\lambda\mathsf{Y}(\mathcal{A}))$ consisting of the following reduction rules:

(1) $\mathbf{R}_i\mathbf{c}_{\vec{a}} \to \mathbf{true}_0$ for $\vec{a} \in R_i$.

(2) $\mathbf{R}_i\mathbf{c}_{\vec{a}} \to \mathbf{false}_0$ for $\vec{a} \in A^{p_i} - R_i$.

(3) $\mathbf{F}_j\mathbf{c}_{\vec{a}} \to \mathbf{c}_{f_j(\vec{a})}$ for $\vec{a} \in A^{q_j}$.

(iii) Let γ be the reductor on $\mathcal{T}(\lambda\mathsf{Y}(\mathcal{A}))$ consisting of the following reduction rules:

$$\mathbf{Y}_\sigma \to \lambda f.f(\mathbf{Y}_\sigma f)$$

with $f \in Var_{\sigma\to\sigma}$.

(iv) $\lambda\mathsf{Y}(\mathcal{A}) = (\mathcal{T}(\lambda\mathsf{Y}(\mathcal{A})), \alpha\beta\gamma)$.

(v) $\lambda\mathsf{Y}\eta(\mathcal{A}) = (\mathcal{T}(\lambda\mathsf{Y}(\mathcal{A})), \alpha\beta\gamma\eta)$. $\square$

1.3.10. Definition. (i) $\mathcal{N} = (\omega, =, S^+)$.

(ii) The constant that represents the equality relation in $\lambda\mathsf{Y}(\mathcal{N})$ will be denoted by $\mathbf{\Delta}$.

(iii) The constant that represents the successor function in $\lambda\mathsf{Y}(\mathcal{N})$ will be denoted by $\mathbf{S}^+$. $\square$

1.3.11. Lemma. (i) $\mathbf{\Delta}\mathbf{c}_m\mathbf{c}_m \to_\alpha \mathbf{true}_0$.

(ii) *If $m \neq n$, then* $\mathbf{\Delta}\mathbf{c}_m\mathbf{c}_n \to_\alpha \mathbf{false}_0$.

(iii) $\mathbf{S}^+\mathbf{c}_m \to_\alpha \mathbf{c}_{m+1}$.

Proof. Simple. $\square$

The systems $\lambda\mathsf{Y}(\mathcal{N})$ and $\lambda\mathsf{Y}\eta(\mathcal{N})$ will be of special interest. The system $\lambda\mathsf{Y}(\mathcal{N})$ is very similar to PCF; see [Plotkin 1977].

In the following we will look at conditions on δ-reduction under which $\beta\delta$-reduction and $\beta\eta\delta$-reduction have the **CR**-property. In particular γ-reduction and $\alpha\gamma$-reduction will satisfy these conditions, so that we have $\mathbf{CR}(\beta\gamma)$, $\mathbf{CR}(\beta\eta\gamma)$, $\mathbf{CR}(\alpha\beta\gamma)$ and $\mathbf{CR}(\alpha\beta\gamma\eta)$.

1.3.12. Definition. Let $\mathbf{R}_1$ and $\mathbf{R}_2$ be reductors on $\mathcal{T}(\mathbb{C})$. Then $\mathbf{R}_1$ commutes with $\mathbf{R}_2$ if for all $M, M_1, M_2 \in \mathcal{T}(\mathbb{C})$ with $M \twoheadrightarrow_{R_1} M_1$ and $M \twoheadrightarrow_{R_2} M_2$ there exists $N \in \mathcal{T}(\mathbb{C})$ such that $M_1 \twoheadrightarrow_{R_2} N$ and $M_2 \twoheadrightarrow_{R_1} N$. $\square$

$$\begin{tikzcd}
M \arrow[r, two heads, "R_1"] \arrow[d, two heads, "R_2"'] & M_1 \arrow[d, two heads, "R_2"] \\
M_2 \arrow[r, two heads, "R_1"'] & N
\end{tikzcd}$$

Note that $\mathbf{R}$ commutes with itself if and only if $\mathbf{CR}(\mathbf{R})$. So self-commutation coincides with the Church-Rosser property.

1.3.13. LEMMA (Hindley-Rosen Lemma). *For every $i \in I$ let $\mathbf{R}_i$ be a reductor on $\mathcal{T}(\mathbb{C})$. If for all $i, j \in I$ $\mathbf{R}_i$ commutes with $\mathbf{R}_j$, then the reductor*

$$\bigcup_{i \in I} \mathbf{R}_i$$

has the $\mathbf{CR}$*-property.*

PROOF. See for example [Klop 1980] or [Barendregt 1984]. □

1.3.14. THEOREM. (i) $\mathbf{CR}(\beta)$.
  (ii) $\mathbf{CR}(\eta)$.
  (iii) *β commutes with η.*
  (iv) $\mathbf{CR}(\beta\eta)$.

PROOF. The proofs of $\mathbf{CR}(\beta)$, $\mathbf{CR}(\eta)$ and the fact that β and η commute are similar to the proofs of the corresponding facts in the type-free λ-calculus; see for example [Barendregt 1984]. Finally, $\mathbf{CR}(\beta\eta)$ follows by the Hindley-Rosen Lemma. □

Let δ be a δ-reductor on $\mathcal{T}(\mathbb{C})$.

1.3.15. DEFINITION. The reductor δ is called closed if every δ-redex is closed. □

Note that δ is closed if and only if for every δ-redex $\mathbf{C}M_1 \ldots M_m$ we have that the arguments $M_1, \ldots, M_m$ are closed. Also note that if δ is closed, every contractum of a δ-redex is also closed.

1.3.16. LEMMA. (i) *Suppose δ is closed. Then*

$$M_1 \rightarrow_\delta M_2 \Rightarrow M_1[x := N] \rightarrow_\delta M_2[x := N].$$

(ii) $N_1 \rightarrow_\delta N_2 \Rightarrow M[x := N_1] \twoheadrightarrow_\delta M[x := N_2]$.

PROOF. (i) By induction on the structure of M_1.

(1) $M_1 \equiv x$. Then $\neg(M_1 \rightarrow_\delta M_2)$.

(2) $M_1 \equiv \mathbf{C}$. Suppose $M_1 \rightarrow_\delta M_2$. Note that M_1 is closed, so M_2 is also closed, and so $M_1[x := N] \equiv M_1 \rightarrow_\delta M_2 \equiv M_2[x := N]$.

(3) $M_1 \equiv PQ$. Suppose $M_1 \rightarrow_\delta^O M_2$ with $O = (C, \Delta)$. Then there are three possibilities.

- $P \to_\delta P'$ and $M_2 \equiv P'Q$. (Then the redex Δ is in P, and $C = C_1 Q$ for some context C_1.) By induction $P[x := N] \to_\delta P'[x := N]$ and so by the compatibility of $\to_\delta$ it follows that $M_1[x := N] \to_\delta M_2[x := N]$.
- $Q \to_\delta Q'$ and $M_2 \equiv PQ'$. (In this case $C = PC_1$ for some C_1, so Δ is in Q.) This case is similar to the previous case.
- $M_1 \equiv \Delta$. Then M_1 is closed, because δ is closed, so M_2 is also closed and we have $M_1[x := N] \equiv M_1 \to_\delta M_2 \equiv M_2[x := N]$.

(4) $M_1 \equiv \lambda y.P$. Suppose $M_1 \to_\delta M_2$. Then $P \to_\delta P'$ and $M_2 \equiv \lambda y.P'$, for some P'. By induction $P[x := N] \to_\delta P'[x := N]$ and so $M_1[x := N] \equiv \lambda y.P[x := N] \to_\delta \lambda y.P'[x := N] \equiv M_2[x := N]$.

(ii) By induction on the structure of M, using the compatibility of $\twoheadrightarrow_\delta$. □

1.3.17. Definition. Let $\mathbf{R}$ be a reductor on $\mathcal{T}(\mathbb{C})$. Then δ is $\mathbf{R}$-normal if for every δ-redex $\mathbf{C}M_1 \ldots M_m$ the arguments $M_1, \ldots, M_m$ are R-normal forms. □

1.3.18. Proposition. (i) *If δ is closed and β-normal, then β commutes with δ.*

(ii) *If δ is closed and η-normal, then η commutes with δ.*

(iii) *If δ is deterministic and δ-normal, then $\mathbf{CR}(\delta)$.*

Proof. (i) Suppose δ is closed and β-normal. Let $M \to_\beta^{O_1} M_1$ and $M \to_\delta^{O_2} M_2$ with $O_1 = (C_1, (\lambda x.P)Q)$ and $O_2 = (C_2, \mathbf{C}N_1 \ldots N_m)$. Then O_1 is not contained in O_2 because $N_1, \ldots, N_m$ are β-normal forms. So the following cases remain.

- O_1 and O_2 are disjoint. Then there trivially exists $N \in \mathcal{T}(\mathbb{C})$ such that $M_1 \to_\delta N$ and $M_2 \to_\beta N$.
- $O_2 \subseteq P$. Say $P \to_\delta P'$ by contraction of $\mathbf{C}N_1 \ldots N_m$ in P. Because δ is closed, the previous lemma gives $P[x := Q] \to_\delta P'[x := Q]$ and so

$$\begin{tikzcd}
M \equiv C_1[(\lambda x.P)Q] \arrow[r, "\delta"] \arrow[d, "\beta"'] & C_1[(\lambda x.P')Q] \equiv M_2 \arrow[d, "\beta"] \\
M_1 \equiv C_1[P[x := Q]] \arrow[r, "\delta"'] & C_1[P'[x := Q]]
\end{tikzcd}$$

- $O_2 \subseteq Q$. Say $Q \to_\delta Q'$ by contracting $\mathbf{C}N_1 \ldots N_m$ in Q. By the previous lemma we have $P[x := Q] \twoheadrightarrow_\delta P[x := Q']$, so

$$\begin{tikzcd}
M \equiv C_1[(\lambda x.P)Q] \arrow[r, "\delta"] \arrow[d, "\beta"'] & C_1[(\lambda x.P)Q'] \equiv M_2 \arrow[d, "\beta"] \\
M_1 \equiv C_1[P[x := Q]] \arrow[r, two heads, "\delta"] & C_1[P[x := Q']]
\end{tikzcd}$$

Therefore we have that if $M \to_\beta M_1$ and $M \to_\delta M_2$ then there exists $N \in \mathcal{T}(\mathbb{C})$ such that $M_1 \twoheadrightarrow_\delta N$ and $M_2 \to_\beta N$. From this it follows that β and δ commute.

(ii) Suppose δ is closed and η-normal. Let $M \to_\eta^{O_1} M_1$ and $M \to_\delta^{O_2} M_2$ with $O_1 = (C_1, \lambda x.Px)$ and $O_2 = (C_2, \mathbf{C}N_1 \ldots N_m)$. Then $N_1, \ldots, N_m$ are closed η-normal forms, so O_1 is not contained in O_2. If O_1 and O_2 are disjoint, then there trivially exists $N \in \mathcal{T}(\mathbb{C})$ such that $M_1 \to_\delta N$ and $M_2 \to_\eta N$. So we may assume $O_2 \subseteq O_1$. Then $\mathbf{C}N_1 \ldots N_m \subseteq Px$, but since N_m is closed, we have $\mathbf{C}N_1 \ldots N_m \subseteq P$. Say $P \to_\delta P'$ by contracting $\mathbf{C}N_1 \ldots N_m$. Then $x \notin FV(P')$ because $x \notin FV(P)$ and so

$$\begin{tikzcd}
M \equiv C_1[\lambda x.Px] \arrow[r, "\delta"] \arrow[d, "\eta"'] & C_1[\lambda x.P'x] \equiv M_2 \arrow[d, "\eta"] \\
M_1 \equiv C_1[P] \arrow[r, "\delta"'] & C_1[P']
\end{tikzcd}$$

So for all $M, M_1, M_2 \in \mathcal{T}(\mathbb{C})$ if $M \to_\eta M_1$ and $M \to_\delta M_2$ then there exists $N \in \mathcal{T}(\mathbb{C})$ such that $M_1 \to_\delta N$ and $M_2 \to_\eta N$. Therefore δ commutes with η.

(iii) Suppose δ is deterministic and δ-normal. Let $M \to_\delta^{O_1} M_1$ and $M \to_\delta^{O_2} M_2$ with $O_1 = (C_1, \mathbf{C}_1 P_1 \ldots P_m)$ and $O_2 = (C_2, \mathbf{C}_2 Q_1 \ldots Q_n)$. We claim that O_1 and O_2 are either disjoint or identical. For suppose that one is included in the other, for example $O_1 \subseteq O_2$. Because δ is δ-normal, every Q_i is a δ-normal form, so O_1 is not contained in one of the Q_i's. Therefore $m \leq n$ and $\mathbf{C}_1 P_1 \ldots P_m \equiv \mathbf{C}_2 Q_1 \ldots Q_m$. But then $\mathbf{C}_1 \equiv \mathbf{C}_2$ and so $m = arity(\mathbf{C}_1) = arity(\mathbf{C}_2) = n$ and therefore $\mathbf{C}_1 P_1 \ldots P_m \equiv \mathbf{C}_2 Q_1 \ldots Q_n$, from which it follows that $O_1 = O_2$. Therefore we have the following two cases.

- O_1 and O_2 are disjoint. Then there trivially exists $N \in \mathcal{T}(\mathbb{C})$ such that $M_1 \to_\delta N$ and $M_2 \to_\delta N$.
- $O_1 = O_2$. Because δ is deterministic it follows that $M_1 \equiv M_2$.

We see that for all $M, M_1, M_2 \in \mathcal{T}(\mathbb{C})$ if $M \to_\delta M_1$ and $M \to_\delta M_2$, then there exists $N \in \mathcal{T}(\mathbb{C})$ such that $M_1 \to_\delta N$ and $M_2 \to_\delta N$. So δ has the Church-Rosser property. □

1.3.19. Theorem (First Church-Rosser Theorem).
Let δ be a δ-reductor on $\mathcal{T}(\mathbb{C})$, such that δ is deterministic and closed.

(i) *If δ is $\beta\delta$-normal, then* $\mathbf{CR}(\beta\delta)$.

(ii) *If δ is $\beta\eta\delta$-normal, then* $\mathbf{CR}(\beta\eta\delta)$.

Proof. By 1.3.14, 1.3.18 and the Hindley-Rosen Lemma. □

In section 5.2 we will prove the Second Church-Rosser Theorem which is a strenghtening of the First Church-Rosser Theorem.

1.3.20. THEOREM. *Let $\mathcal{A}$ be a structure.*
(i) $\lambda\mathsf{Y}(\mathcal{A}) \models \mathbf{CR}$.
(ii) $\lambda\mathsf{Y}\eta(\mathcal{A}) \models \mathbf{CR}$.

PROOF. By the First Church-Rosser Theorem, using that $\alpha\gamma$ is deterministic, closed and $\alpha\beta\gamma\eta$-normal. $\square$

1.3.21. COROLLARY. (i) $\lambda\mathsf{Y}(\mathcal{N}) \models \mathbf{CR}$.
(ii) $\lambda\mathsf{Y}\eta(\mathcal{N}) \models \mathbf{CR}$. $\square$

1.4. Computability.

A measure for the strength of a system is the number of functions that can be computed in it.

1.4.1. DEFINITION. Let Σ be a system. A numeral system in Σ is a sequence

$$\mathbf{a} = (\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2, \ldots)$$

in $\mathcal{T}_0(\Sigma)^0$, such that for all $m, n \in \omega$

$$\Sigma \vdash \mathbf{a}_m = \mathbf{a}_n \Rightarrow m = n.$$

If $m \in \omega$, then $\mathbf{a}_m$ is called the numeral of m. $\square$

1.4.2. DEFINITION. Let Σ be a system and let $\mathbf{a} = (\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2, \ldots)$ be a numeral system in Σ. Let $k \in \omega$ and $f : \omega^k \to \omega$.
(i) Let $F \in \mathcal{T}_{t(k)}(\Sigma)^0$. Then F Σ-computes f w.r.t. $\mathbf{a}$ if

$$\Sigma \vdash F\mathbf{a}_{m_1}\mathbf{a}_{m_2} \ldots \mathbf{a}_{m_k} = \mathbf{a}_{f(m_1,\ldots,m_k)}$$

for all $(m_1, \ldots, m_k) \in \omega^k$.
(ii) f is Σ-computable w.r.t. $\mathbf{a}$ if there exists $F \in \mathcal{T}_{t(k)}(\Sigma)^0$ such that F Σ-computes f w.r.t. $\mathbf{a}$. $\square$

Let $\mathbf{c}$ be the sequence $(\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2, \ldots)$ of constants in $\lambda\mathsf{Y}(\mathcal{N})$ that represent the natural numbers. Because $\mathbf{CR}(\alpha\beta\gamma)$ and every $\mathbf{c}_m$ is an $\alpha\beta\gamma$-normal form, we have that for all $m, n \in \omega$

$$\lambda\mathsf{Y}(\mathcal{N}) \vdash \mathbf{c}_m = \mathbf{c}_n \Rightarrow m = n$$

so $\mathbf{c}$ is a numeral system in $\lambda\mathsf{Y}(\mathcal{N})$.

In the following it will be shown that all recursive functions on ω are $\lambda\mathsf{Y}(\mathcal{N})$-computable w.r.t. $\mathbf{c}$. First we will give a definition of the class *REC* of all recursive functions.

1.4.3. Definition. The initial functions are the functions Z,S^+ and U_i^k defined as follows.

(i) The constant-0 function $Z : \omega \to \omega$ is defined by $Z(m) = 0$.

(ii) The successor function $S^+ : \omega \to \omega$ is defined by $S^+(m) = m + 1$.

(iii) Let $k, i \in \omega$ with $1 \leq i \leq k$. The i-th k-ary projection function $U_i^k : \omega^k \to \omega$ is defined by $U_i^k(m_1, \ldots, m_k) = m_i$. □

1.4.4. Definition. Let $f : \omega^k \to \omega$ and let $g_1, \ldots, g_k : \omega^l \to \omega$. The composition of f and $g_1, \ldots, g_k$ is the function $h : \omega^l \to \omega$ defined by

$$h(\vec{m}) = f(g_1(\vec{m}), \ldots, g_k(\vec{m})).$$

Notation: $h = Comp[f, g_1, \ldots, g_k]$. □

1.4.5. Definition. Let $f : \omega^k \to \omega$ and $g : \omega^{k+2} \to \omega$. The primitive recursion of f and g is the function $h : \omega^{k+1} \to \omega$ defined by

$$\begin{array}{rcl} h(\vec{m}, 0) & = & f(\vec{m}) \\ h(\vec{m}, n+1) & = & g(h(\vec{m}, n), \vec{m}, n). \end{array}$$

Notation: $h = Prim[f, g]$. □

1.4.6. Definition. (i) Let $R \subseteq \omega^{k+1}$. Then R is regular if

$$\forall \vec{m} \in \omega^k \exists n \in \omega[R(\vec{m}, n)].$$

(ii) Let $R \subseteq \omega^{k+1}$ be regular and let $\vec{m} \in \omega^k$. Then $\mu n[R(\vec{m}, n)]$ denotes the least $n \in \omega$ such that $R(\vec{m}, n)$ holds.

(iii) Let $f : \omega^{k+1} \to \omega$. Then f is called regular if

$$\forall \vec{m} \in \omega^k \exists n \in \omega[f(\vec{m}, n) = 0]$$

that is, if the set $\{(\vec{m}, n) | f(\vec{m}, n) = 0\}$ is regular. □

1.4.7. Definition. Let $f : \omega^{k+1} \to \omega$ be regular. The unbounded search of f is the function $g : \omega^k \to \omega$ defined by

$$g(\vec{m}) = \mu n[f(\vec{m}, n) = 0].$$

Notation: $g = Us[f]$. □

1.4.8. Definition. The class REC of all recursive functions is the least class of functions on ω that contains all initial functions and is closed under composition, primitive recursion and (regular) unbounded search. □

It will now be shown that $\lambda\mathsf{Y}(\mathcal{N})$ computes every recursive function w.r.t. $\mathbf{c}$. From now on we will call a function $\lambda\mathsf{Y}(\mathcal{N})$-computable if it is $\lambda\mathsf{Y}(\mathcal{N})$-computable w.r.t. $\mathbf{c}$.

1.4.9. LEMMA. *The initial functions are $\lambda\mathsf{Y}(\mathcal{N})$-computable.*

PROOF. The term $\mathsf{Z} \equiv \lambda x.\mathbf{c}_0$ computes Z, the term $\mathbf{S}^+$ computes S^+ and for $k, i \in \omega$ with $1 \leq i \leq k$ the term $\mathsf{U}_i^k \equiv \lambda x_1 \dots x_k.x_i$ computes U_i^k. $\square$

1.4.10. LEMMA. *The class of $\lambda\mathsf{Y}(\mathcal{N})$-computable functions is closed under composition.*

PROOF. Simple. $\square$

1.4.11. LEMMA. *The class of $\lambda\mathsf{Y}(\mathcal{N})$-computable functions is closed under (regular) unbounded search.*

PROOF. Let $f : \omega^{k+1} \to \omega$ be regular and suppose that f is $\lambda\mathsf{Y}(\mathcal{N})$-computable. Let $g = Us[f] : \omega^k \to \omega$. Let $F \in \mathcal{T}_{t(k+1)}(\lambda\mathsf{Y}(\mathcal{N}))^0$ $\lambda\mathsf{Y}(\mathcal{N})$-compute f. Define

$$H \equiv \mathbf{Y}_{t(k+1)}(\lambda h\vec{x}y.\mathbf{if}\ \mathbf{\Delta}(F\vec{x}y)\mathbf{c}_0\ \mathbf{then}\ y\ \mathbf{else}\ h\vec{x}(\mathbf{S}^+y)).$$

Then $H =_{\alpha\beta\gamma} \lambda\vec{x}y.\mathbf{if}\ \mathbf{\Delta}(F\vec{x}y)\mathbf{c}_0\ \mathbf{then}\ y\ \mathbf{else}\ H\vec{x}(\mathbf{S}^+y)$. And so for all $\vec{m} \in \omega^k$ and $n \in \omega$

(i) $f(\vec{m}, n) = 0 \Rightarrow H\mathbf{c}_{\vec{m}}\mathbf{c}_n =_{\alpha\beta\gamma} \mathbf{c}_n$.

(ii) $f(\vec{m}, n) > 0 \Rightarrow H\mathbf{c}_{\vec{m}}\mathbf{c}_n =_{\alpha\beta\gamma} H\mathbf{c}_{\vec{m}}\mathbf{c}_{n+1}$.

Now define $G \equiv \lambda\vec{x}.H\vec{x}\mathbf{c}_0$. Then G $\lambda\mathsf{Y}(\mathcal{N})$-computes g. $\square$

1.4.12. LEMMA. *The predecessor function $P^- : \omega \to \omega$ defined by $P^-(0) = 0$ and $P^-(m+1) = m$ is $\lambda\mathsf{Y}(\mathcal{N})$-computable.*

PROOF. For every $m \in \omega$ we have

$$P^-(m) = \mu n[m = 0 \vee m = S^+(n)].$$

So $P^- = Us[f]$ where $f : \omega^2 \to \omega$ is defined by $f(m, n) = 0$ if $m = 0 \vee m = S^+(n)$ and $f(m, n) = 1$ otherwise. Then f is $\lambda\mathsf{Y}(\mathcal{N})$-computable, and by the previous lemma it follows that P^- is $\lambda\mathsf{Y}(\mathcal{N})$-computable. $\square$

1.4.13. LEMMA. *The class of $\lambda\mathsf{Y}(\mathcal{N})$-computable functions is closed under primitive recursion.*

PROOF. Let P^- $\lambda\mathsf{Y}(\mathcal{N})$-compute P^-. Let $f : \omega^k \to \omega$ and $g : \omega^{k+2} \to \omega$ be $\lambda\mathsf{Y}(\mathcal{N})$-computable. Let $h = Prim[f, g]$. Let F $\lambda\mathsf{Y}(\mathcal{N})$-compute f and let G $\lambda\mathsf{Y}(\mathcal{N})$-compute g. Define

$$H \equiv \mathbf{Y}_{t(k+1)}(\lambda h\vec{x}y.\mathbf{if}\ \mathbf{\Delta}y\mathbf{c}_0\ \mathbf{then}\ F\vec{x}\ \mathbf{else}\ G(h\vec{x}(\mathsf{P}^-y))\vec{x}(\mathsf{P}^-y)).$$

Then H $\lambda\mathsf{Y}(\mathcal{N})$-computes h. $\square$

1.4.14. THEOREM. *Every recursive function is $\lambda\mathsf{Y}(\mathcal{N})$-computable.*

PROOF. By 1.4.9, 1.4.10, 1.4.11 and 1.4.13. □

The converse of this result also holds (that is, every $\lambda\mathsf{Y}(\mathcal{N})$-computable function is recursive), and there also exists a notion of $\lambda\mathsf{Y}(\mathcal{N})$-computability for partial functions on ω, such that the $\lambda\mathsf{Y}(\mathcal{N})$-computable partial functions are precisely the partial recursive functions. We will not go into this, because all we need to know in the following is that every recursive function is $\lambda\mathsf{Y}(\mathcal{N})$-computable.

1.5. Translations.

A translation from a system Σ_1 into a system Σ_2 is a mapping from $\mathcal{T}(\Sigma_1)$ into $\mathcal{T}(\Sigma_2)$ that preserves conversion.

Let $\mathbb{C}^1 = (\mathbb{C}^1_\sigma)_{\sigma\in\mathbb{T}}$ and $\mathbb{C}^2 = (\mathbb{C}^2_\sigma)_{\sigma\in\mathbb{T}}$ be two constant families. Then we have two sets $\mathcal{T}(\mathbb{C}^1)$ and $\mathcal{T}(\mathbb{C}^2)$ of typed λ-terms.

1.5.1. DEFINITION. A transformation from $\mathcal{T}(\mathbb{C}^1)$ into $\mathcal{T}(\mathbb{C}^2)$ is a mapping $\Xi : \mathcal{T}(\mathbb{C}^1) \to \mathcal{T}(\mathbb{C}^2)$ such that

(i) $\mathbf{C} \in \mathbb{C}^1_\sigma \Rightarrow \Xi(\mathbf{C}) \in \mathcal{T}_\sigma(\mathbb{C}^2)^0$.
(ii) $\Xi(x) \equiv x$.
(iii) $\Xi(MN) \equiv \Xi(M)\Xi(N)$.
(iv) $\Xi(\lambda x.M) \equiv \lambda x.\Xi(M)$. □

1.5.2. LEMMA. *Let $\Xi : \mathcal{T}(\mathbb{C}^1) \to \mathcal{T}(\mathbb{C}^2)$ be a transformation.*

(i) $M \in \mathcal{T}_\sigma(\mathbb{C}^1) \Rightarrow \Xi(M) \in \mathcal{T}_\sigma(\mathbb{C}^2)$.
(ii) $FV(\Xi(M)) = FV(M)$.
(iii) $\Xi(M[x := N]) \equiv \Xi(M)[x := \Xi(N)]$.
(iv) $M \equiv N \Rightarrow \Xi(M) \equiv \Xi(N)$.

PROOF. Simple. □

1.5.3. LEMMA. *Let Ξ_1 and Ξ_2 be two transformations from $\mathcal{T}(\mathbb{C}^1)$ into $\mathcal{T}(\mathbb{C}^2)$ such that for all $\mathbf{C} \in |\mathbb{C}^1|$ we have $\Xi_1(\mathbf{C}) \equiv \Xi_2(\mathbf{C})$. Then for all $M \in \mathcal{T}(\mathbb{C}^1)$ we have $\Xi_1(M) \equiv \Xi_2(M)$.*

PROOF. Simple, by induction on the structure of M. □

Let Σ_1 and Σ_2 be two systems.

1.5.4. DEFINITION. (i) A translation from Σ_1 into Σ_2 is a transformation $\Xi : \mathcal{T}(\Sigma_1) \to \mathcal{T}(\Sigma_2)$ such that

$$\Sigma_1 \vdash M = N \Rightarrow \Sigma_2 \vdash \Xi(M) = \Xi(N)$$

for all $M, N \in \mathcal{T}(\Sigma_1)$. If Ξ is a translation from Σ_1 into Σ_2, then we write $\Xi : \Sigma_1 \to \Sigma_2$.

(ii) Σ_1 is translatable into Σ_2 (notation $\Sigma_1 \trianglelefteq \Sigma_2$) if there exists a translation $\Xi : \Sigma_1 \to \Sigma_2$. □

1.5.5. LEMMA. *Let $\Xi : \mathcal{T}(\Sigma_1) \to \mathcal{T}(\Sigma_2)$ be a transformation such that for every Σ_1-axiom $(M = N)$ we have $\Sigma_2 \vdash \Xi(M) = \Xi(N)$. Then $\Xi : \Sigma_1 \to \Sigma_2$.*

PROOF. Simple. □

If Σ_1 is a $\beta\delta$-system and Σ_2 is a β-system, then Σ_1 is translatable into Σ_2 if for every constant $\mathbf{C}$ of Σ_1 we can find a closed Σ_2-term that 'behaves' like $\mathbf{C}$.

1.5.6. DEFINITION. Let $\xi : |\mathbb{C}^1| \to \mathcal{T}(\mathbb{C}^2)^0$ be type-preserving: $\mathbf{C} \in \mathbb{C}^1_\sigma \Rightarrow \xi(\mathbf{C}) \in \mathcal{T}_\sigma(\mathbb{C}^2)^0$. Define $\Xi_\xi : \mathcal{T}(\mathbb{C}^1) \to \mathcal{T}(\mathbb{C}^2)$ as follows.

(i) $\Xi_\xi(\mathbf{C}) \equiv \xi(\mathbf{C})$.

(ii) $\Xi_\xi(x) \equiv x$.

(iii) $\Xi_\xi(MN) \equiv \Xi_\xi(M)\Xi_\xi(N)$.

(iv) $\Xi_\xi(\lambda x.M) \equiv \lambda x.\Xi_\xi(M)$. □

1.5.7. LEMMA. *If $\xi : |\mathbb{C}^1| \to \mathcal{T}(\mathbb{C}^2)^0$ is type-preserving, then Ξ_ξ is a transformation from $\mathcal{T}(\mathbb{C}^1)$ into $\mathcal{T}(\mathbb{C}^2)$.*

PROOF. Trivial. □

1.5.8. PROPOSITION. *Let Σ_1 be a $\beta\delta$-system, let Σ_2 be a β-system and let $\Xi : \mathcal{T}(\Sigma_1) \to \mathcal{T}(\Sigma_2)$ be a transformation such that for every δ-axiom $(M = N)$ of Σ_1 we have $\Sigma_2 \vdash \Xi(M) = \Xi(N)$. Then $\Xi : \Sigma_1 \to \Sigma_2$.*

PROOF. By lemma 1.5.5 we only need to prove that

$$\Sigma_2 \vdash \Xi((\lambda x.M)N) = \Xi(M[x := N])$$

but this follows by 1.5.2. □

1.5.9. THEOREM. $\lambda\mathsf{T} \trianglelefteq \lambda\mathsf{Y}\eta(\mathcal{N})$.

PROOF. First we need $\lambda\mathsf{Y}\eta(\mathcal{N})$-terms that decide equality on ω using booleans of type $\sigma \to \sigma \to \sigma$ for any $\sigma \in \mathbb{T}$. Define $\mathbf{\Delta}_\sigma \equiv \lambda abxy\vec{z}.\mathbf{\Delta}ab(x\vec{z})(y\vec{z})$ such that $\mathbf{\Delta}_\sigma$ is of type $0 \to 0 \to \sigma \to \sigma \to \sigma$. Then for all $m, n \in \omega$

(i) $\mathbf{\Delta}_\sigma \mathbf{c}_m \mathbf{c}_n =_{\alpha\beta\gamma\eta} \mathbf{true}_\sigma$ if $m = n$

(ii) $\mathbf{\Delta}_\sigma \mathbf{c}_m \mathbf{c}_n =_{\alpha\beta\gamma\eta} \mathbf{false}_\sigma$ if $m \neq n$.
Using the $\mathbf{\Delta}_\sigma$'s we can construct primitive recursors R_σ in $\lambda\mathsf{Y}\eta(\mathcal{N})$ as follows. Define

$$\mathsf{R}_\sigma \equiv \mathbf{Y}_\tau(\lambda rxyz.\mathbf{if}\ \mathbf{\Delta}_\sigma z\mathbf{c}_0\ \mathbf{then}\ x\ \mathbf{else}\ y(rxy(\mathsf{P}^- z))(\mathsf{P}^- z))$$

with $\tau = \sigma \to (\sigma \to 0 \to \sigma) \to 0 \to \sigma$. Then for all $M \in \mathcal{T}_\sigma(\lambda\mathsf{Y}\eta(\mathcal{N})), N \in \mathcal{T}_{\sigma\to 0\to\sigma}(\lambda\mathsf{Y}\eta(\mathcal{N}))$ and $m \in \omega$

(i) $\mathsf{R}_\sigma MN\mathbf{c}_0 =_{\alpha\beta\gamma\eta} M$

(ii) $\mathsf{R}_\sigma MN\mathbf{c}_{m+1} =_{\alpha\beta\gamma\eta} N(\mathsf{R}_\sigma MN\mathbf{c}_m)\mathbf{c}_m$.

Let $\mathbb{C}$ be such that $\mathcal{T}(\mathbb{C}) = \mathcal{T}(\lambda\mathsf{T})$ and define $\xi : |\mathbb{C}| \to \mathcal{T}(\lambda\mathsf{Y}\eta(\mathcal{N}))^0$ by

(i) $\xi(\mathbf{c}_m) \equiv \mathbf{c}_m$.

(ii) $\xi(\mathbf{S}^+) \equiv \mathbf{S}^+$.

(iii) $\xi(\mathbf{R}_\sigma) \equiv \mathsf{R}_\sigma$.

Then $\Xi_\xi : \lambda\mathsf{T} \to \lambda\mathsf{Y}\eta(\mathcal{N})$ by proposition 1.5.8. □

Between translations and computability we have the following connection.

1.5.10. PROPOSITION. *Let $\Xi : \Sigma_1 \to \Sigma_2$. Let $\mathbf{a} = (\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2, \ldots)$ be a numeral system in Σ_1 and let $\mathbf{b} = (\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_2, \ldots)$ be a numeral system in Σ_2, such that $\Xi(\mathbf{a}_m) \equiv \mathbf{b}_m$ for every $m \in \omega$. Then every function on ω that is Σ_1-computable w.r.t. $\mathbf{a}$ is Σ_2-computable w.r.t. $\mathbf{b}$.*

PROOF. Simple. □

1.6. Simply typed combinatory logic.

Combinatory logic is a variant of λ-calculus without bound variables. Combinatory terms are built from constants and variables using application only. Combinatory logic is sometimes useful in the construction of models for simply typed λ-calculi.

1.6.1. DEFINITION. Let $\mathbb{C} = (\mathbb{C}_\sigma)_{\sigma\in\mathbb{T}}$ be a constant family.

(i) For every $\sigma \in \mathbb{T}$ the set $\mathbf{CL}_\sigma(\mathbb{C})$ of all combinatory terms of type σ with constants from $\mathbb{C}$ is defined inductively as follows.

(1) $\mathbb{C}_\sigma \subseteq \mathbf{CL}_\sigma(\mathbb{C})$.

(2) $Var_\sigma \subseteq \mathbf{CL}_\sigma(\mathbb{C})$.

(3) $P \in \mathbf{CL}_{\sigma\to\tau}(\mathbb{C})\ \&\ Q \in \mathbf{CL}_\sigma(\mathbb{C}) \Rightarrow (PQ) \in \mathbf{CL}_\tau(\mathbb{C})$.

(ii) The set of all combinatory terms with constants from $\mathbb{C}$:

$$\mathbf{CL}(\mathbb{C}) = \bigcup_{\sigma\in\mathbb{T}} \mathbf{CL}_\sigma(\mathbb{C}).\ \Box$$

For every $\sigma \in \mathbb{T}$ we have $\mathbf{CL}_\sigma(\mathbb{C}) \subseteq \mathcal{T}_\sigma(\mathbb{C})$, and also $\mathbf{CL}(\mathbb{C}) \subseteq \mathcal{T}(\mathbb{C})$.

1.6.2. DEFINITION. (i) For every $P \in \mathbf{CL}(\mathbb{C})$ the set $FV(P)$ of all (free) variables of P is defined by induction on the structure of P as follows.

(1) $FV(\mathbf{C}) = \emptyset$.

(2) $FV(x) = \{x\}$.

(3) $FV(PQ) = FV(P) \cup FV(Q)$.

(ii) If $FV(P) = \emptyset$, then P is closed. The set of all closed combinatory terms of type σ is denoted by $\mathbf{CL}_\sigma(\mathbb{C})^0$, and the set of all closed combinatory terms is denoted by $\mathbf{CL}(\mathbb{C})^0$.

(iii) Let $\sigma \in \mathbb{T}$, $x \in Var_\sigma$ and $Q \in \mathbf{CL}_\sigma(\mathbb{C})$. For every $P \in \mathbf{CL}(\mathbb{C})$ the substitution $P[x := Q]$ is defined by induction on the structure of P as follows.

(1) $\mathbf{C}[x := Q] = \mathbf{C}$.

(2) $x[x := Q] = Q$.

(3) $y[x := Q] = y$ if $y \neq x$.

(4) $(P_1P_2)[x := Q] = (P_1[x := Q]P_2[x := Q])$. $\Box$

We will now extend $\mathbb{C}$ to a constant family $\mathbb{C}^*$ by adding special constants.

1.6.3. DEFINITION. (i) For all $\sigma, \tau \in \mathbb{T}$ let $\mathsf{K}_{\sigma,\tau}$ be a new constant of type $\sigma \to \tau \to \sigma$.

(ii) For all $\sigma, \tau, \upsilon \in \mathbb{T}$ let $\mathsf{S}_{\sigma,\tau,\upsilon}$ be a new constant of type $(\sigma \to \tau \to \upsilon) \to (\sigma \to \tau) \to \sigma \to \upsilon$.

(iii) For every $\sigma \in \mathbb{T}$ define a set $\mathbb{C}^*_\sigma$ of constants of type σ as follows.

(1) $\mathbb{C}^*_\sigma = \mathbb{C}_\sigma \cup \{\mathsf{K}_{\sigma_0,\sigma_1}\}$ if $\sigma = \sigma_0 \to \sigma_1 \to \sigma_0$.

(2) $\mathbb{C}^*_\sigma = \mathbb{C}_\sigma \cup \{\mathsf{S}_{\sigma_0,\sigma_1,\sigma_2}\}$ if $\sigma = (\sigma_0 \to \sigma_1 \to \sigma_2) \to (\sigma_0 \to \sigma_1) \to \sigma_0 \to \sigma_2$.

(3) $\mathbb{C}^*_\sigma = \mathbb{C}_\sigma$ for all other σ.

(iv) $\mathbb{C}^* = (\mathbb{C}^*_\sigma)_{\sigma \in \mathbb{T}}$. $\Box$

On every set $\mathbf{CL}(\mathbb{C}^*)$ we define weak reduction and weak conversion.

1.6.4. DEFINITION. (i) One-step w-reduction is the binary relation $\to_w$ on $\mathbf{CL}(\mathbb{C}^*)$ which is defined inductively as follows.

(1) $\mathsf{K}_{\sigma,\tau} PQ \to_w P$.

(2) $\mathsf{S}_{\sigma,\tau,\upsilon} PQR \to_w PR(QR)$.

(3) $P \to_w Q \Rightarrow PR \to_w QR$.

(4) $P \to_w Q \Rightarrow RP \to_w RQ$.

(ii) Weak reduction or w-reduction $\twoheadrightarrow_w$ on $\mathbf{CL}(\mathbb{C}^*)$ is the reflexive and transitive closure of $\to_w$.

(iii) Weak conversion or w-conversion $=_w$ on $\mathbf{CL}(\mathbb{C}^*)$ is the symmetric and transitive closure of $\twoheadrightarrow_w$. □

1.6.5. DEFINITION. For every $\sigma \in \mathbb{T}$ define

$$\mathsf{I}_\sigma = \mathsf{S}_{\sigma,\sigma\to\sigma,\sigma}\mathsf{K}_{\sigma,\sigma\to\sigma}\mathsf{K}_{\sigma,\sigma}. \ \square$$

1.6.6. LEMMA. *Let $\sigma \in \mathbb{T}$.*
(i) $\mathsf{I}_\sigma \in \mathbf{CL}_{\sigma\to\sigma}(\mathbb{C}^*)^0$.
(ii) $\mathsf{I}_\sigma P \twoheadrightarrow_w P$.
(iii) $\mathsf{I}_\sigma P =_w P$.

PROOF. Simple. □

Using the K- and S-constants we can simulate abstraction in $\mathbf{CL}(\mathbb{C}^*)$.

1.6.7. DEFINITION. Let $\sigma \in \mathbb{T}$, $x \in Var_\sigma$. For every $P \in \mathbf{CL}_\tau(\mathbb{C}^*)$ we define a term $\lambda^*x.P$ in $\mathbf{CL}_{\sigma\to\tau}(\mathbb{C}^*)$ by induction on the structure of P as follows.
(i) $\lambda^*x.x = \mathsf{I}_\sigma$.
(ii) $\lambda^*x.P = \mathsf{K}_{\tau,\sigma}P$ if $x \notin FV(P)$.
(iii) $\lambda^*x.PQ = \mathsf{S}_{\sigma,\tau,\upsilon}(\lambda^*x.P)(\lambda^*x.Q)$
if $x \in FV(PQ), P \in \mathbf{CL}_{\tau\to\upsilon}(\mathbb{C}^*), Q \in \mathbf{CL}_\tau(\mathbb{C}^*)$. □

1.6.8. LEMMA. (i) $FV(\lambda^*x.P) = FV(P) - \{x\}$.
(ii) $\lambda^*x.P = \lambda^*y.P[x := y]$ *if* $y \notin FV(P)$.
(iii) $(\lambda^*x.P)[x := Q] = \lambda^*x.P[x := Q]$ *if* $x \neq y$ *and* $x \notin FV(Q)$.
(iv) $(\lambda^*x.P)Q \twoheadrightarrow_w P[x := Q]$.

PROOF. Easy, by induction on the structure of P. □

Using λ^*-abstraction it is possible to construct a transformation mapping from $\mathcal{T}(\mathbb{C})$ into $\mathbf{CL}(\mathbb{C}^*)$.

1.6.9. DEFINITION. (i) The map $CL : \mathcal{T}(\mathbb{C}) \to \mathbf{CL}(\mathbb{C}^*)$ is defined by induction on the structure of the terms in $\mathcal{T}(\mathbb{C})$ as follows.

(1) $CL(\mathbf{C}) = \mathbf{C}$.

(2) $CL(x) = x$.

(3) $CL(MN) = CL(M)CL(N)$.

(4) $CL(\lambda x.M) = \lambda^*x.CL(M)$.

(ii) Instead of $CL(M)$ we will sometimes write $(M)_{CL}$ or M_{CL}. □

1.6.10. LEMMA. (i) $M \in \mathcal{T}_\sigma(\mathbb{C}) \Rightarrow CL(M) \in \mathbf{CL}_\sigma(\mathbb{C}^*)$.
(ii) $M \equiv N \Rightarrow CL(M) = CL(N)$.

(iii) $FV(M_{CL}) = FV(M)$.
(iv) $CL(M[x := N]) = CL(M)[x := CL(N)]$.
(v) $CL((\lambda x.M)N) \twoheadrightarrow_w CL(M[x := N])$.

PROOF. Simple. □

In general we do not have that

$$M =_\beta N \Rightarrow CL(M) =_w CL(N)$$

because in general we do not have

$$P =_w Q \Rightarrow \lambda^* x.P =_w \lambda^* x.Q.$$

A counterexample is provided by the fact that $\lambda^* x.\mathsf{I}_\sigma \neq_w \lambda^* x.x$.

Chapter 2

Models

This chapter deals with the model theory of simply typed λ-calculi. The basic structures in which simply typed λ-terms can be interpreted are type structures, so type structures are the potential models for simply typed λ-calculi. In interpreting an equational theory such as a simply typed λ-calculus in a type structure there are (at least) two possible interpretations for equality. The first alternative is interpreting equality in the system by equality in type structures. This leads to algebras and models for systems. The second alternative is interpreting equality in a system by extensional equality in type structures. This leads to quasi-algebras and quasi-models for systems.

2.1. Type structures.

A type structure consists of a collection of typed objects together with an operation called application. In a type structure objects of type $\sigma \to \tau$ can be applied to objects of type σ yielding objects of type τ.

2.1.1. Definition. (i) A type structure is a structure

$$\mathfrak{M} = ((\mathfrak{M}_\sigma)_{\sigma \in \mathbb{T}}, (App_{\sigma,\tau})_{\sigma,\tau \in \mathbb{T}})$$

such that for all $\sigma, \tau \in \mathbb{T}$

(1) $\mathfrak{M}_\sigma$ is a non-empty set

(2) $App_{\sigma,\tau} : \mathfrak{M}_{\sigma\to\tau} \times \mathfrak{M}_\sigma \to \mathfrak{M}_\tau$.

(ii) The elements of $\mathfrak{M}_\sigma$ are the type-σ objects of $\mathfrak{M}$.

(iii) The functions $App_{\sigma,\tau}$ are called application functions.

(iv) The domain of $\mathfrak{M}$ is defined by

$$|\mathfrak{M}| = \bigcup_{\sigma \in \mathbb{T}} \mathfrak{M}_\sigma. \ \square$$

Notation. (i) $App_{\sigma,\tau}(a, b) = ab$.

(ii) For application in a type structure we use association to the left, so $ab_1 \ldots b_n$ denotes $(\ldots((ab_1)b_2)\ldots b_{n-1})b_n$. $\square$

In a few moments it will be shown that every type structure is isomorphic with a type structure in which every object has a unique type. Therefore the notation ab (although it is ambiguous) will not cause any problems.

In the following $\mathfrak{M}, \mathfrak{M}^1, \mathfrak{M}^2, \mathfrak{M}^3, \ldots$ are type structures, with

$$\mathfrak{M} = ((\mathfrak{M}_\sigma)_{\sigma \in \mathbb{T}}, (App_{\sigma,\tau})_{\sigma,\tau \in \mathbb{T}})$$

and for $n = 1, 2, 3, \ldots$

$$\mathfrak{M}^n = ((\mathfrak{M}^n_\sigma)_{\sigma \in \mathbb{T}}, (App^n_{\sigma,\tau})_{\sigma,\tau \in \mathbb{T}}).$$

Our first examples of type structures are the full type structures over (non-empty) sets.

2.1.2. Definition. Let A be a non-empty set.

(i) For every $\sigma \in \mathbb{T}$ define A_σ by induction on the structure of σ as follows.

(1) $A_0 = A$.

(2) $A_{\sigma \to \tau} = (A_\tau)^{A_\sigma}$.

(ii) Define $App^A_{\sigma,\tau} : A_{\sigma\to\tau} \times A_\sigma \to A_\tau$ to be the usual function application:

$$App^A_{\sigma,\tau}(f, a) = f(a).$$

(iii) The full type structure over A is defined by

$$\mathfrak{M}(A) = ((A_\sigma)_{\sigma \in \mathbb{T}}, (App^A_{\sigma,\tau})_{\sigma,\tau \in \mathbb{T}}). \ \square$$

2.1.3. Definition. A homomorphism from $\mathfrak{M}^1$ into $\mathfrak{M}^2$ is a family $f = (f_\sigma)_{\sigma \in \mathbb{T}}$ of mappings $f_\sigma : \mathfrak{M}^1_\sigma \to \mathfrak{M}^2_\sigma$ such that for all $\sigma, \tau \in \mathbb{T}$, $a \in \mathfrak{M}^1_{\sigma\to\tau}$, $b \in \mathfrak{M}^1_\sigma$

$$f_{\sigma\to\tau}(a) f_\sigma(b) = f_\tau(ab).$$

If f is a homomorphism from $\mathfrak{M}^1$ into $\mathfrak{M}^2$, then we write $f : \mathfrak{M}^1 \to \mathfrak{M}^2$. Instead of $f_\sigma(a)$ we often write $f(a)$. $\square$

2.1.4. Definition. Let $f : \mathfrak{M}^1 \to \mathfrak{M}^2$ with $f = (f_\sigma)_{\sigma \in \mathbb{T}}$.

(i) f is injective (notation $f : \mathfrak{M}^1 \rightarrowtail \mathfrak{M}^2$) if f_σ is injective for every $\sigma \in \mathbb{T}$. An injective homomorphism is called an embedding.

(ii) f is surjective (notation $f : \mathfrak{M}^1 \twoheadrightarrow \mathfrak{M}^2$) if f_σ is surjective for every $\sigma \in \mathbb{T}$. A surjective homomorphism is called a covering.

(iii) f is bijective (notation $f : \mathfrak{M}^1 \rightarrowtail\!\!\!\!\twoheadrightarrow \mathfrak{M}^2$) if f_σ is bijective for every $\sigma \in \mathbb{T}$. A bijective homomorphism is called an isomorphism. □

Note that a homomorphism is bijective if and only if it is injective and surjective.

2.1.5. DEFINITION. The type structures $\mathfrak{M}^1$ and $\mathfrak{M}^2$ are isomorphic (notation $\mathfrak{M}^1 \cong \mathfrak{M}^2$) if there exists an isomorphism from $\mathfrak{M}^1$ into $\mathfrak{M}^2$. □

Next it will be shown that every type structure is isomorphic with a type structure in which every object has a unique type.

2.1.6. DEFINITION. (i) $\mathfrak{M}$ is separated if for all $\sigma, \tau \in \mathbb{T}$

$$\sigma \neq \tau \Rightarrow \mathfrak{M}_\sigma \cap \mathfrak{M}_\tau = \emptyset.$$

(ii) The separation of $\mathfrak{M}$ is the type structure

$$\mathfrak{M}^S = ((\mathfrak{M}^S_\sigma)_{\sigma \in \mathbb{T}}, (App^S_{\sigma,\tau})_{\sigma,\tau \in \mathbb{T}})$$

with for all $\sigma, \tau \in \mathbb{T}$

(1) $\mathfrak{M}^S_\sigma = \mathfrak{M}_\sigma \times \{\sigma\} = \{(a, \sigma) | a \in \mathfrak{M}_\sigma\}$

(2) $App^S_{\sigma,\tau} : \mathfrak{M}^S_{\sigma\to\tau} \times \mathfrak{M}^S_\sigma \to \mathfrak{M}^S_\tau$ is defined by

$$App^S_{\sigma,\tau}((a, \sigma\to\tau), (b, \sigma)) = (ab, \tau). \ \square$$

2.1.7. LEMMA. (i) $\mathfrak{M} \cong \mathfrak{M}^S$.
(ii) $\mathfrak{M}^S$ *is separated.*

PROOF. (i) Let $f = (f_\sigma)_{\sigma\in\mathbb{T}}$ with $f_\sigma : \mathfrak{M}_\sigma \to \mathfrak{M}^S_\sigma$ defined by $f_\sigma(a) = (a, \sigma)$. Then $f : \mathfrak{M} \to \mathfrak{M}^S$ is an isomorphism.
(ii) Trivial. □

From now on we will assume every type structure that we work with to be separated.

Next we will look at mappings and functionals on type structures.

2.1.8. DEFINITION. Let $\sigma, \tau \in \mathbb{T}$ with $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0$.
(i) $(\mathfrak{M})_\sigma = \mathfrak{M}_{\sigma_1} \times \mathfrak{M}_{\sigma_2} \times \ldots \times \mathfrak{M}_{\sigma_k}$.
(ii) $\sigma * \tau = \sigma_1 \to \ldots \to \sigma_k \to \tau$. □

2.1.9. LEMMA. *Let* $\sigma, \tau \in \mathbb{T}$.

(i) *If* $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0$ *and* $\tau = \tau_1 \to \ldots \to \tau_l \to 0$, *then*

$$\sigma * \tau = \sigma_1 \to \ldots \to \sigma_k \to \tau_1 \to \ldots \to \tau_l \to 0.$$

(ii) $(\mathfrak{M})_{\sigma*\tau} = (\mathfrak{M})_\sigma \times (\mathfrak{M})_\tau$.
(iii) $(\mathfrak{M})_0 = \{()\}$ *where* $()$ *is the empty sequence.*
(iv) $(\mathfrak{M})_{t(k)} = (\mathfrak{M}_0)^k$ *for every* $k \in \omega$.
(v) *The set of all finite sequences of objects of* $\mathfrak{M}$ *is*

$$|\mathfrak{M}|^* = \bigcup_{\sigma \in \mathbb{T}} (\mathfrak{M})_\sigma.$$

Proof. Trivial. □

2.1.10. Definition. (i) A mapping on $\mathfrak{M}$ is a map $\varphi : (\mathfrak{M})_\sigma \to \mathfrak{M}_\tau$. The type $\sigma * \tau$ is called the type of φ and the number $fac(\sigma)$ is called the arity of φ.
(ii) A functional on $\mathfrak{M}$ is a mapping $\varphi : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$. □

Notation. For mappings and functionals on type structures we use $\varphi, \psi, \chi, \ldots$ □

2.1.11. Examples. Let $\sigma \in \mathbb{T}$.
(i) The application functional $\mathsf{Ap}_\sigma : \mathfrak{M}_\sigma \times (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ is defined by

$$\mathsf{Ap}_\sigma(a, \vec{b}) = a\vec{b}.$$

(ii) For every $m \in \mathfrak{M}_0$ the constant-m functional of type σ is the functional $\mathsf{C}^\sigma_m : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ defined by

$$\mathsf{C}^\sigma_m(\vec{a}) = m. \ \Box$$

2.1.12. Definition. Let $\sigma, \tau \in \mathbb{T}$, $\varphi : (\mathfrak{M})_\sigma \to \mathfrak{M}_\tau$.
(i) Let $a \in \mathfrak{M}_{\sigma*\tau}$. Then a $\mathfrak{M}$-represents φ if

$$\forall \vec{b} \in (\mathfrak{M})_\sigma [\, \varphi(\vec{b}) = a\vec{b}\,].$$

(ii) φ is $\mathfrak{M}$-representable if there exists an object in $\mathfrak{M}_{\sigma*\tau}$ that $\mathfrak{M}$-represents φ. □

We will be especially interested in $\mathfrak{M}$-representable functionals on $\mathfrak{M}$.

2.1.13. Definition. (i) For every $\sigma \in \mathbb{T}$ the set of all $\mathfrak{M}$-representable functionals $\varphi : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ is denoted by $\mathbb{F}_\sigma(\mathfrak{M})$.
(ii) The set of all $\mathfrak{M}$-representable functionals:

$$\mathbb{F}(\mathfrak{M}) = \bigcup_{\sigma \in \mathbb{T}} \mathbb{F}_\sigma(\mathfrak{M}).$$

(iii) The function $fnl : |\mathfrak{M}| \to \mathbb{F}(\mathfrak{M})$ is defined as follows. Let $\sigma \in \mathbb{T}$, $a \in \mathfrak{M}_\sigma$. Then $fnl(a) : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ is defined by $fnl(a)(\vec{b}) = a\vec{b}$. $\square$

If two objects (of the same type) have the same applicative behaviour (after descending all the way down to type 0), then they are called extensionally equal.

2.1.14. Definition. (i) Extensional equality is the relation $\approx$ on $\mathfrak{M}$ defined as follows. For all $\sigma \in \mathbb{T}$ and $a, b \in \mathfrak{M}_\sigma$ define

$$a \approx b \iff \forall \vec{c} \in (\mathfrak{M})_\sigma [\, a\vec{c} = b\vec{c}\,].$$

If $a \approx b$, then a and b are called extensionally equal.

(ii) Let $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0 \in \mathbb{T}$, $\vec{a} = (a_1, \ldots, a_k) \in (\mathfrak{M})_\sigma$ and $\vec{b} = (b_1, \ldots, b_k) \in (\mathfrak{M})_\sigma$. Then

$$\vec{a} \approx \vec{b} \iff a_i \approx b_i \; (i = 1, \ldots, k). \; \square$$

2.1.15. Definition. (i) $\mathfrak{M}$ is extensional if for all $\sigma \in \mathbb{T}$, $a, b \in \mathfrak{M}_\sigma$

$$a \approx b \Rightarrow a = b.$$

(ii) $\mathfrak{M}$ is quasi-extensional if for all $\sigma, \tau \in \mathbb{T}$, $a, a' \in \mathfrak{M}_{\sigma\to\tau}$, $b, b' \in \mathfrak{M}_\sigma$

$$a \approx a' \;\&\; b \approx b' \Rightarrow ab \approx a'b'. \; \square$$

2.1.16. Lemma. *Every extensional type structure is quasi-extensional, but not every quasi-extensional type structure is extensional.*

Proof. Simple. $\square$

If $\mathfrak{M}$ is quasi-extensional, then the quotient structure of $\mathfrak{M}$ w.r.t. extensional equality is an extensional type structure $\mathfrak{M}^E$.

2.1.17. Definition. Let $\mathfrak{M}$ be quasi-extensional.

(i) For all $\sigma \in \mathbb{T}$ and $a \in \mathfrak{M}_\sigma$ define

$$[a] = \{b \in \mathfrak{M}_\sigma \mid a \approx b\}.$$

(ii) For every $\sigma \in \mathbb{T}$

$$\mathfrak{M}^E_\sigma = \{[a] \mid a \in \mathfrak{M}_\sigma\}.$$

(iii) Define $App^E_{\sigma,\tau} : \mathfrak{M}^E_{\sigma\to\tau} \times \mathfrak{M}^E_\sigma \to \mathfrak{M}^E_\tau$ by

$$App^E_{\sigma,\tau}([a], [b]) = [ab].$$

By quasi-extensionality $App^E_{\sigma,\tau}$ is well-defined.

(iv) The extensional quotient of $\mathfrak{M}$ is defined by

$$\mathfrak{M}^E = ((\mathfrak{M}^E_\sigma)_{\sigma\in\mathbb{T}}, (App^E_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}). \ \square$$

2.1.18. PROPOSITION. *Let $\mathfrak{M}$ be quasi-extensional. Then $\mathfrak{M}^E$ is extensional.*

PROOF. Let $\approx^E$ denote extensional equality in $\mathfrak{M}^E$. Let $\sigma \in \mathbb{T}$ and let $a, b \in \mathfrak{M}_\sigma$ be such that $[a] \approx^E [b]$. Then for all $\vec{c} \in (\mathfrak{M})_\sigma$

$$[a\vec{c}] = [a][\vec{c}] = [b][\vec{c}] = [b\vec{c}]$$

so $a\vec{c} = b\vec{c}$. So $a \approx b$ and therefore $[a] = [b]$. $\square$

In the following we will take a look at a few characterizations of quasi-extensionality and of extensionality.

2.1.19. DEFINITION. (i) Let $\sigma \in \mathbb{T}$, $a \in \mathfrak{M}_\sigma$. Then a is $\mathfrak{M}$-extensional if for all $\vec{b}, \vec{c} \in (\mathfrak{M})_\sigma$

$$\vec{b} \approx \vec{c} \Rightarrow a\vec{b} = a\vec{c}.$$

(ii) $\mathfrak{M}$ is locally extensional if every object in $\mathfrak{M}$ is $\mathfrak{M}$-extensional. $\square$

2.1.20. LEMMA. *Let $\sigma, \tau \in \mathbb{T}$.*

(i) *For all $a \in \mathfrak{M}_{\sigma\to\tau}$, $b, c \in \mathfrak{M}_\sigma$: if a is $\mathfrak{M}$-extensional and $b \approx c$, then $ab \approx ac$.*

(ii) *For all $a, b \in \mathfrak{M}_{\sigma\to\tau}$, $c \in \mathfrak{M}_\sigma$*

$$a \approx b \Rightarrow ac \approx bc.$$

PROOF. (i) Let $a \in \mathfrak{M}_{\sigma\to\tau}$ be $\mathfrak{M}$-extensional and let $b, c \in \mathfrak{M}_\sigma$ be such that $b \approx c$. Let $\vec{d} \in (\mathfrak{M})_\tau$. Then $(b, \vec{d}) \approx (c, \vec{d})$, so $ab\vec{d} = ac\vec{d}$. So $ab \approx ac$.

(ii) Let $a, b \in \mathfrak{M}_{\sigma\to\tau}$ be such that $a \approx b$ and let $c \in \mathfrak{M}_\sigma$. Let $\vec{d} \in (\mathfrak{M})_\tau$. Then $(c, \vec{d}) \in (\mathfrak{M})_{\sigma\to\tau}$, so $ac\vec{d} = bc\vec{d}$. So $ac \approx bc$. $\square$

2.1.21. THEOREM. *$\mathfrak{M}$ is quasi-extensional if and only if $\mathfrak{M}$ is locally extensional.*

PROOF. (i) Suppose $\mathfrak{M}$ is quasi-extensional. Let $\sigma \in \mathbb{T}$, $a \in \mathfrak{M}_\sigma$. Let $\vec{b}, \vec{c} \in (\mathfrak{M})_\sigma$ be such that $\vec{b} \approx \vec{c}$. Using quasi-extensionality it follows by induction on m, that

$$\forall m \leq fac(\sigma)[\, ab_1 \ldots b_m \approx ac_1 \ldots c_m \,].$$

So $a\vec{b} \approx a\vec{c}$, so $a\vec{b} = a\vec{c}$. So a is $\mathfrak{M}$-extensional. Therefore $\mathfrak{M}$ is locally extensional.

(ii) Now suppose $\mathfrak{M}$ is locally extensional. Let $\sigma, \tau \in \mathbb{T}$, $a, a' \in \mathfrak{M}_{\sigma\to\tau}$, $b, b' \in \mathfrak{M}_\sigma$ be such that $a \approx a'$ and $b \approx b'$. Because a is $\mathfrak{M}$-extensional $ab \approx ab'$. And because $a \approx a'$ we have $ab' \approx a'b'$. So $ab \approx a'b'$. Therefore $\mathfrak{M}$ is quasi-extensional. $\square$

2.1.22. THEOREM. *$\mathfrak{M}$ is extensional if and only if for all $\sigma, \tau \in \mathbb{T}$ and $a, b \in \mathfrak{M}_{\sigma\to\tau}$*

$$\forall c \in \mathfrak{M}_\sigma[\, ac = bc\,] \Rightarrow a = b.$$

PROOF. (i) Suppose $\mathfrak{M}$ is extensional. Then for all $\sigma, \tau \in \mathbb{T}$ and $a, b \in \mathfrak{M}_{\sigma\to\tau}$

$$\forall c \in \mathfrak{M}_\sigma[\, ac = bc\,] \Rightarrow a \approx b \Rightarrow a = b.$$

(ii) Now assume that for all $\sigma, \tau \in \mathbb{T}$ and $a, b \in \mathfrak{M}_{\sigma\to\tau}$

$$\forall c \in \mathfrak{M}_\sigma[\, ac = bc\,] \Rightarrow a = b.$$

For all $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0 \in \mathbb{T}$, $m \leq k$ and $a, b \in \mathfrak{M}_\sigma$ define

$$a =_m b$$

if for all $c_1 \in \mathfrak{M}_{\sigma_1}, \ldots, c_m \in \mathfrak{M}_{\sigma_m}$

$$ac_1 \ldots c_m = bc_1 \ldots c_m.$$

Using the assumption it can be shown that

$$\forall m < k[\, a =_{m+1} b \Rightarrow a =_m b\,].$$

But $a \approx b \Leftrightarrow a =_k b$ and $a = b \Leftrightarrow a =_0 b$. So for all a and b

$$a \approx b \Rightarrow a =_k b \Rightarrow a =_0 b \Rightarrow a = b$$

and therefore $\mathfrak{M}$ is extensional. $\square$

Next we will give a new characterization of extensionality using topology.

2.1.23. DEFINITION. Let $\sigma \in \mathbb{T}$.

(i) For all $a, b \in \mathfrak{M}_\sigma$ and $A \subseteq (\mathfrak{M})_\sigma$ define

$$a =_A b \Longleftrightarrow \forall \vec{c} \in A[\, a\vec{c} = b\vec{c}\,].$$

(ii) A subset U of $\mathfrak{M}_\sigma$ is called open if for every $a \in U$ there exists a finite $A \subseteq (\mathfrak{M})_\sigma$ such that for all $b \in \mathfrak{M}_\sigma$

$$a =_A b \Rightarrow b \in U.$$

(iii) The set of all open subsets of $\mathfrak{M}_\sigma$ is denoted by θ_σ. $\square$

2.1.24. PROPOSITION. *θ_σ is a topology on $\mathfrak{M}_\sigma$ for every $\sigma \in \mathbb{T}$.*

PROOF. Let $\sigma \in \mathbb{T}$.

(i) $\emptyset$ and $\mathfrak{M}_\sigma$ are open.

(ii) Let $(U_i)_{i\in I}$ be a family of open subsets of $\mathfrak{M}_\sigma$, and let $U = \bigcup\{U_i|\ i \in I\}$. Let $a \in U$. Then $a \in U_{i_0}$ for some $i_0 \in I$. Because U_{i_0} is open there exists a finite $A \subseteq (\mathfrak{M})_\sigma$ such that for all $b \in \mathfrak{M}_\sigma$

$$a =_A b \Rightarrow b \in U_{i_0}.$$

But then for all $b \in \mathfrak{M}_\sigma$

$$a =_A b \Rightarrow b \in U.$$

Therefore U is open.

(iii) Let $U, V \in \theta_\sigma$. Let $a \in U \cap V$. Then there exist finite $A, B \subseteq (\mathfrak{M})_\sigma$ such that for all $b \in \mathfrak{M}_\sigma$

$$a =_A b \Rightarrow b \in U$$

$$a =_B b \Rightarrow b \in V.$$

Now $A \cup B$ is finite, and for all $b \in \mathfrak{M}_\sigma$

$$a =_{A\cup B} b \Rightarrow b \in U \cap V.$$

So $U \cap V$ is open. □

If θ is a topology on a set X, then $a, b \in X$ are called θ-separable if there exist disjoint $U, V \in \theta$ such that $a \in U$ and $b \in V$. Two points of X are called θ-inseparable if they are not θ-separable. Further θ is said to have the Hausdorff Property if for all $a, b \in X$ with $a \neq b$ we have that a and b are θ-separable.

2.1.25. Definition. Let $\sigma \in \mathbb{T}$.

(i) A finite type-σ functional on $\mathfrak{M}$ is a mapping $\varphi : A \to \mathfrak{M}_0$ such that A is a finite subset of $(\mathfrak{M})_\sigma$.

(ii) Let $\varphi : A \to \mathfrak{M}_0$ be a finite type-σ functional on $\mathfrak{M}$ and let $a \in \mathfrak{M}_\sigma$. Then

$$\varphi \preceq a \iff \forall \vec{b} \in A[\,\varphi(\vec{b}) = a\vec{b}\,].$$

(iii) For every finite type-σ functional on $\mathfrak{M}$:

$$B_\sigma(\varphi) = \{a \in \mathfrak{M}_\sigma \,|\, \varphi \preceq a\}.\ \square$$

2.1.26. Lemma. *If $\sigma \in \mathbb{T}$ and φ is a finite type-σ functional on $\mathfrak{M}$, then $B_\sigma(\varphi)$ is an open subset of $\mathfrak{M}_\sigma$.*

Proof. Simple. □

2.1.27. Proposition. *Let $\sigma \in \mathbb{T}$, $a, b \in \mathfrak{M}_\sigma$. Then $a \approx b$ if and only if a and b are θ_σ-inseparable.*

PROOF. (i) Let $a \approx b$. Suppose a and b are θ_σ-separable. Then there exist disjoint $U, V \in \theta_\sigma$ such that $a \in U$ and $b \in V$. Let A be a finite subset of $(\mathfrak{M})_\sigma$ such that for all $c \in \mathfrak{M}_\sigma$

$$a =_A c \Rightarrow c \in U.$$

Because $a \approx b$ we have $a =_A b$ and so $b \in U$. Contradiction.

(ii) Let $a \not\approx b$. Then $a\vec{c} \neq b\vec{c}$ for some $\vec{c} \in (\mathfrak{M})_\sigma$. Define $A = \{\vec{c}\}$ and define $\varphi, \psi : A \to \mathfrak{M}_0$ by $\varphi(\vec{c}) = a\vec{c}$ and $\psi(\vec{c}) = b\vec{c}$. Then $B_\sigma(\varphi) \cap B_\sigma(\psi) = \emptyset$, $a \in B_\sigma(\varphi)$ and $b \in B_\sigma(\psi)$, so a and b are θ_σ-separable. □

2.1.28. THEOREM. *$\mathfrak{M}$ is extensional if and only if θ_σ has the Hausdorff Property for every $\sigma \in \mathbb{T}$.*

PROOF. By 2.1.27. □

2.2. Σ-Algebras and Σ-models.

In this section we will define the notions of Σ-interpretation, Σ-algebra and Σ-model. For every one of these notions we will also define a quasi-variant, so we also define quasi-Σ-interpretation, quasi-Σ-algebra and quasi-Σ-model. These 'quasi-notions' arise naturally in working with quasi-extensional type structures. In chapter 4 quasi-$\lambda\mathsf{Y}(\mathcal{N})$-models will arise naturally in working with recursion theories on quasi-extensional type structures.

2.2.1. DEFINITION. (i) A valuation in $\mathfrak{M}$ is a map $\rho : Var_{\mathbb{T}} \to |\mathfrak{M}|$ such that for all $\sigma \in \mathbb{T}$ and $x \in Var_\sigma$ we have $\rho(x) \in \mathfrak{M}_\sigma$.

(ii) The set of all valuations in $\mathfrak{M}$ is denoted by $Val(\mathfrak{M})$.

(iii) Let $\rho \in Val(\mathfrak{M}), k \in \omega, \sigma_1, \ldots, \sigma_k \in \mathbb{T}, \vec{x} = x_1, \ldots, x_k \in Var_{\mathbb{T}}$, pairwise distinct, with $x_i \in Var_{\sigma_i}$ and $\vec{a} = a_1, \ldots, a_k$ with $a_i \in \mathfrak{M}_{\sigma_i}$. Then $\rho(\vec{x} := \vec{a}) \in Val(\mathfrak{M})$ is defined by

(1) $\rho(\vec{x} := \vec{a})(x_i) = a_i$ for $i = 1, \ldots, k$).

(2) $\rho(\vec{x} := \vec{a})(x) = \rho(x)$ if $x \notin \{x_1, \ldots, x_k\}$.

(iv) For all $\rho_1, \rho_2 \in Val(\mathfrak{M})$ and $M \in \mathcal{T}(\mathbb{C})$

$$\rho_1 =_M \rho_2 \Longleftrightarrow \forall x \in FV(M)[\rho_1(x) = \rho_2(x)]. \ \square$$

2.2.2. DEFINITION. (i) A quasi-$\mathcal{T}(\mathbb{C})$-interpretation in $\mathfrak{M}$ is a mapping

$$[\![\,]\!] : \mathcal{T}(\mathbb{C}) \times Val(\mathfrak{M}) \to |\mathfrak{M}|$$

$$(M, \rho) \mapsto [\![M]\!]_\rho$$

such that the following hold.

(1) $M \in \mathcal{T}_\sigma(\mathbb{C}) \Rightarrow [\![M]\!]_\rho \in \mathfrak{M}_\sigma$.

(2) $M \equiv N \Rightarrow [\![M]\!]_\rho = [\![N]\!]_\rho$.

(3) $[\![x]\!]_\rho = \rho(x)$.

(4) $[\![MN]\!]_\rho = [\![M]\!]_\rho [\![N]\!]_\rho$.

(5) $[\![\lambda x.M]\!]_\rho a \approx [\![M]\!]_{\rho(x:=a)}$.

(6) $\rho_1 =_M \rho_2 \Rightarrow [\![M]\!]_{\rho_1} = [\![M]\!]_{\rho_2}$.

(Note the extensional equality in (5).)

(ii) A $\mathcal{T}(\mathbb{C})$-interpretation in $\mathfrak{M}$ is a quasi-$\mathcal{T}(\mathbb{C})$-interpretation $[\![\,]\!]$ in $\mathfrak{M}$ such that

$$[\![\lambda x.M]\!]_\rho a = [\![M]\!]_{\rho(x:=a)}.$$

(iii) If $[\![\,]\!]$ is a quasi-$\mathcal{T}(\mathbb{C})$-interpretation in $\mathfrak{M}$, then $(\mathfrak{M}, [\![\,]\!])$ is called a quasi-$\mathcal{T}(\mathbb{C})$-structure, and if $[\![\,]\!]$ is a $\mathcal{T}(\mathbb{C})$-interpretation in $\mathfrak{M}$, then $(\mathfrak{M}, [\![\,]\!])$ is called a $\mathcal{T}(\mathbb{C})$-structure.

(iv) Let Σ be a system with $\mathcal{T}(\Sigma) = \mathcal{T}(\mathbb{C})$. A quasi-$\Sigma$-interpretation is a quasi-$\mathcal{T}(\mathbb{C})$-interpretation and a quasi-Σ-structure is a quasi-$\mathcal{T}(\mathbb{C})$-structure. A Σ-interpretation is a $\mathcal{T}(\mathbb{C})$-interpretation and a Σ-structure is a $\mathcal{T}(\mathbb{C})$-structure. □

If $[\![\,]\!]$ is a quasi-Σ-interpretation and $M \in \mathcal{T}_\sigma(\Sigma)^0$, then $[\![M]\!]_\rho$ is independent of ρ and will often be denoted by $[\![M]\!]$.

2.2.3. Definition. Let $(\mathfrak{M}, [\![\,]\!])$ be a quasi-Σ-structure.

(i) Quasi-satisfaction in $(\mathfrak{M}, [\![\,]\!])$ is defined by

$$\mathfrak{M}, [\![\,]\!] \models_q M = N \iff \forall \rho \in Val(\mathfrak{M})[[\![M]\!]_\rho \approx [\![N]\!]_\rho].$$

(ii) The set of quasi-theorems of $(\mathfrak{M}, [\![\,]\!])$ is defined by

$$Th_q(\mathfrak{M}, [\![\,]\!]) = \{(M = N) | \mathfrak{M}, [\![\,]\!] \models_q M = N\}.$$

(iii) $(\mathfrak{M}, [\![\,]\!])$ is a quasi-Σ-algebra if for all $M, N \in \mathcal{T}(\Sigma)$

$$\Sigma \vdash M = N \Rightarrow \mathfrak{M}, [\![\,]\!] \models_q M = N.$$

(iv) $(\mathfrak{M}, [\![\,]\!])$ is a quasi-Σ-model if $Th_q(\mathfrak{M}, [\![\,]\!])$ is a Σ-theory. □

2.2.4. Definition. Let $(\mathfrak{M}, [\![\,]\!])$ be a Σ-structure.

(i) Satisfaction in $(\mathfrak{M}, [\![\,]\!])$ is defined by

$$\mathfrak{M}, [\![\,]\!] \models M = N \iff \forall \rho \in Val(\mathfrak{M})[[\![M]\!]_\rho = [\![N]\!]_\rho].$$

(ii) The set of theorems of $(\mathfrak{M}, [\![\,]\!])$ is defined by

$$Th(\mathfrak{M}, [\![\,]\!]) = \{(M = N) | \mathfrak{M}, [\![\,]\!] \models M = N\}.$$

(iii) $(\mathfrak{M}, [\![\,]\!])$ is a Σ-algebra if for all $M, N \in \mathcal{T}(\Sigma)$

$$\Sigma \vdash M = N \Rightarrow \mathfrak{M}, [\![\,]\!] \models M = N.$$

(iv) $(\mathfrak{M}, [\![\,]\!])$ is a Σ-model if $Th(\mathfrak{M}, [\![\,]\!])$ is a Σ-theory. □

2.2.5. LEMMA. (i) *Every quasi-Σ-model is a quasi-Σ-algebra.*
(ii) *Every Σ-model is a Σ-algebra.*
(iii) *Every Σ-algebra is a quasi-Σ-algebra.*
(iv) *Every Σ-model is a quasi-Σ-algebra.*

PROOF. (i) $Th(\Sigma)$ is the least Σ-theory.
(ii) $Th(\Sigma)$ is the least Σ-theory.
(iii) By the fact that $Th(\mathfrak{M}, [\![\,]\!]) \subseteq Th_q(\mathfrak{M}, [\![\,]\!])$.
(iv) By (ii) and (iii). □

In section 2.6 it will be shown that no other inclusions than these hold in general between the classes of quasi-Σ-algebras, quasi-Σ-models, Σ-algebras and Σ-models. In particular it will be shown that not every model is a quasi-model.

2.2.6. LEMMA. *Let $(\mathfrak{M}, [\![\,]\!])$ be a quasi-Σ-structure such that $\mathfrak{M}$ is extensional.*

(i) *$[\![\,]\!]$ is a Σ-interpretation.*
(ii) *$Th_q(\mathfrak{M}, [\![\,]\!]) = Th(\mathfrak{M}, [\![\,]\!])$.*
(iii) *If $(\mathfrak{M}, [\![\,]\!])$ is a quasi-Σ-algebra, then $(\mathfrak{M}, [\![\,]\!])$ is a Σ-algebra.*
(iv) *If $(\mathfrak{M}, [\![\,]\!])$ is a quasi-Σ-model, then $(\mathfrak{M}, [\![\,]\!])$ is a Σ-model.*

PROOF. Simple. □

2.2.7. PROPOSITION. *Let $(\mathfrak{M}, [\![\,]\!])$ be a quasi-Σ-structure.*
(i) *$Th_q(\mathfrak{M}, [\![\,]\!])$ is a theory if and only if for all $\sigma, \tau \in \mathbb{T}, M, N \in \mathcal{T}_\sigma(\Sigma), L \in \mathcal{T}_{\sigma\to\tau}(\Sigma)$*

$$\mathfrak{M}, [\![\,]\!] \models_q M = N \Rightarrow \mathfrak{M}, [\![\,]\!] \models_q LM = LN.$$

(ii) *If $\mathfrak{M}$ is quasi-extensional, then $Th_q(\mathfrak{M}, [\![\,]\!])$ is a theory.*

PROOF. (i) By 1.2.25.
(ii) By (i). □

2.2.8. PROPOSITION. *Let $(\mathfrak{M}, [\![\,]\!])$ be a Σ-structure.*
(i) *$Th(\mathfrak{M}, [\![\,]\!])$ is a theory if and only if for all $\sigma, \tau \in \mathbb{T}, x \in Var_\sigma, M, N \in \mathcal{T}_\tau(\Sigma)$*

$$\mathfrak{M}, [\![\,]\!] \models M = N \Rightarrow \mathfrak{M}, [\![\,]\!] \models \lambda x.M = \lambda x.N.$$

(ii) *If $\mathfrak{M}$ is extensional, then $Th(\mathfrak{M}, [\![\,]\!])$ is a theory.*

PROOF. (i) By 1.2.25.

(ii) We can give two proofs for this. First proof: by (i). Second proof: by 2.2.6 and 2.2.7. □

2.2.9. THEOREM. *Let* $(\mathfrak{M}, [\![\,]\!])$ *be a quasi-*Σ*-structure, such that* $\mathfrak{M}$ *is quasi-extensional and* $Ax(\Sigma) \subseteq Th_q(\mathfrak{M}, [\![\,]\!])$*. Then* $(\mathfrak{M}, [\![\,]\!])$ *is a quasi-*Σ*-model.*

PROOF. By 2.2.7. □

2.2.10. THEOREM. *Let* $(\mathfrak{M}, [\![\,]\!])$ *be a* Σ*-structure, such that* $\mathfrak{M}$ *is extensional and* $Ax(\Sigma) \subseteq Th(\mathfrak{M}, [\![\,]\!])$*. Then* $(\mathfrak{M}, [\![\,]\!])$ *is a* Σ*-model.*

PROOF. By 2.2.6 and 2.2.9. □

Next we will look at satisfaction and quasi-satisfaction of β-axioms and η-axioms.

2.2.11. LEMMA (Substitution Lemma).
Let $(\mathfrak{M}, [\![\,]\!])$ *be a quasi-*Σ*-structure such that* $\mathfrak{M}$ *is quasi-extensional. Then for all* $\sigma, \tau \in \mathbb{T}, M \in \mathcal{T}_\sigma(\Sigma), x \in Var_\tau, N \in \mathcal{T}_\tau(\Sigma)$ *and* $\rho \in Val(\mathfrak{M})$

$$[\![M[x := N]]\!]_\rho \approx [\![M]\!]_{\rho(x:=[\![N]\!]_\rho)}.$$

PROOF. By induction on the structure of M. Let $\tau \in \mathbb{T}, x \in Var_\tau, N \in \mathcal{T}_\tau(\Sigma)$. For every $\rho \in Val(\mathfrak{M})$ define $\rho^* = \rho(x := [\![N]\!]_\rho)$. Then for each $y \not\equiv x$ we have $\rho(y := a)^* = \rho^*(y := a)$.

(i) $[\![x[x := N]]\!]_\rho = [\![N]\!]_\rho = \rho^*(x) = [\![x]\!]_{\rho^*}$.

(ii) If $y \not\equiv x$, then $[\![y[x := N]]\!]_\rho = [\![y]\!]_\rho = [\![y]\!]_{\rho^*}$ because $\rho =_y \rho^*$.

(iii) $[\![\mathbf{C}[x := N]]\!]_\rho = [\![\mathbf{C}]\!]_\rho = [\![\mathbf{C}]\!]_{\rho^*}$.

(iv) Let $M \equiv M_1 M_2$. Then by induction we have

$$[\![M_1[x := N]]\!]_\rho \approx [\![M_1]\!]_{\rho^*} \;\&\; [\![M_2[x := N]]\!]_\rho \approx [\![M_2]\!]_{\rho^*}$$

so by quasi-extensionality we have

$$\begin{aligned}
[\![(M_1 M_2)[x := N]]\!]_\rho &= [\![M_1[x := N] M_2[x := N]]\!]_\rho \\
&= [\![M_1[x := N]]\!]_\rho [\![M_2[x := N]]\!]_\rho \\
&\approx [\![M_1]\!]_{\rho^*} [\![M_2]\!]_{\rho^*} \\
&= [\![M_1]\!]_{\rho^*} [\![M_2]\!]_{\rho^*}.
\end{aligned}$$

(v) Let $M \equiv \lambda y.M_1$ with $y \in Var_\sigma$. Then for all $a \in \mathfrak{M}_\sigma$

$$\begin{aligned}
[\![(\lambda y.M_1)[x := N]]\!]_\rho a &= [\![\lambda y.M_1[x := N]]\!]_\rho a \\
&\approx [\![M_1[x := N]]\!]_{\rho(y:=a)} \\
&\approx [\![M_1]\!]_{\rho(y:=a)^*} \\
&= [\![M_1]\!]_{\rho^*(y:=a)} \\
&\approx [\![\lambda y.M_1]\!]_{\rho^*} a
\end{aligned}$$

so $[\![(\lambda y.M_1)[x := N]]\!]_\rho \approx [\![\lambda y.M_1]\!]_{\rho^*}$. □

2.2.12. COROLLARY. *Let $(\mathfrak{M}, [\![\ ]\!])$ be a quasi-Σ-structure such that $\mathfrak{M}$ is quasi-extensional. Then*

$$\mathfrak{M}, [\![\ ]\!] \models_q (\lambda x.M)N = M[x := N].$$

PROOF. For all $\rho \in Val(\mathfrak{M})$

$$\begin{array}{rcl} [\![(\lambda x.M)N]\!]_\rho & = & [\![\lambda x.M]\!]_\rho [\![N]\!]_\rho \\ & \approx & [\![M]\!]_{\rho(x:=[\![N]\!]_\rho)} \\ & \approx & [\![M[x := N]]\!]_\rho \end{array}$$

by the Substitution Lemma. □

2.2.13. LEMMA. *Let $(\mathfrak{M}, [\![\ ]\!])$ be a quasi-Σ-structure. If $x \notin FV(M)$, then*

$$\mathfrak{M}, [\![\ ]\!] \models_q \lambda x.Mx = M$$

PROOF. Let $x \in Var_\sigma, \rho \in Val(\mathfrak{M})$. Then for all $a \in \mathfrak{M}_\sigma$

$$\begin{array}{rcl} [\![\lambda x.Mx]\!]_\rho a & \approx & [\![Mx]\!]_{\rho(x:=a)} \\ & = & [\![M]\!]_{\rho(x:=a)} [\![x]\!]_{\rho(x:=a)} \\ & = & [\![M]\!]_\rho a. \end{array}$$

So $[\![\lambda x.Mx]\!]_\rho \approx [\![M]\!]_\rho$ for all ρ. □

2.2.14. THEOREM. *Let Σ be a $\beta\eta\delta$-system and let $(\mathfrak{M}, [\![\ ]\!])$ be a quasi-Σ-structure, such that $\mathfrak{M}$ is quasi-extensional and $Ax_\delta(\Sigma) \subseteq Th_q(\mathfrak{M}, [\![\ ]\!])$. Then $(\mathfrak{M}, [\![\ ]\!])$ is a quasi-Σ-model.*

PROOF. By 2.2.9, 2.2.12 and 2.2.13. □

2.2.15. THEOREM. *Let Σ be a $\beta\eta\delta$-system and let $(\mathfrak{M}, [\![\ ]\!])$ be a Σ-structure, such that $\mathfrak{M}$ is extensional and $Ax_\delta(\Sigma) \subseteq Th(\mathfrak{M}, [\![\ ]\!])$. Then $(\mathfrak{M}, [\![\ ]\!])$ is a Σ-model.*

PROOF. By the previous theorem. □

2.2.16. DEFINITION. (i) The type structure $\mathfrak{M}$ is called a pre-quasi-Σ-algebra resp. pre-quasi-Σ-model if there exists a quasi-Σ-interpretation in $\mathfrak{M}$ such that $(\mathfrak{M}, [\![\ ]\!])$ is a quasi-Σ-algebra resp. quasi-Σ-model.

(ii) The type structure $\mathfrak{M}$ is called a pre-Σ-algebra resp. pre-Σ-model if there exists a Σ-interpretation in $\mathfrak{M}$ such that $(\mathfrak{M}, [\![\ ]\!])$ is a Σ-algebra resp. Σ-model. □

Next it will be shown that every full type structure $\mathfrak{M}(A)$ is a pre-$\lambda^{\rightarrow}\eta(\mathbb{C})$-model, and that the full type structure $\mathfrak{M}(\omega)$ over ω is a pre-$\lambda\mathsf{T}$-model.

2.2.17. DEFINITION. Let A be a non-empty set. Let $\mu : |\mathbb{C}| \to |\mathfrak{M}(A)|$ be type-preserving: $\mathbf{C} \in \mathbb{C}_\sigma \Rightarrow \mu(\mathbf{C}) \in \mathfrak{M}_\sigma$. Now define

$$[\![\;]\!]^\mu : \mathcal{T}(\mathbb{C}) \times Val(\mathfrak{M}(A)) \to |\mathfrak{M}(A)|$$

$$(M, \rho) \mapsto [\![M]\!]^\mu_\rho$$

by induction on the structure of M as follows.

(i) $[\![\mathbf{C}]\!]^\mu_\rho = \mu(\mathbf{C})$.

(ii) $[\![x]\!]^\mu_\rho = \rho(x)$.

(iii) $[\![MN]\!]^\mu_\rho = [\![M]\!]^\mu_\rho [\![N]\!]^\mu_\rho$.

(iv) $[\![\lambda x.M]\!]^\mu_\rho(a) = [\![M]\!]^\mu_{\rho(x:=a)}$. □

2.2.18. THEOREM. *Let A be a non-empty set and let $\mu : |\mathbb{C}| \to |\mathfrak{M}(A)|$ be type-preserving. Then $(\mathfrak{M}(A), [\![\;]\!]^\mu)$ is a $\lambda^\to\eta(\mathbb{C})$-model.*

PROOF. By 2.2.15 and the extensionality of $\mathfrak{M}(A)$. □

2.2.19. THEOREM. *$\mathfrak{M}(\omega)$ is a pre-$\lambda\mathsf{T}$-model.*

PROOF. For every $\sigma \in \mathbb{T}$ define the object $\mathbf{r}_\sigma$ of type $\sigma \to (\sigma \to 0 \to \sigma) \to 0 \to \sigma$ in $\mathfrak{M}(\omega)$ as follows. Let $f \in \omega_\sigma$ and $g \in \omega_{\sigma\to 0\to\sigma}$. Now define $\mathbf{r}_\sigma fg : \omega \to \omega_\sigma$ by

(i) $\mathbf{r}_\sigma fg(0) = f$

(ii) $\mathbf{r}_\sigma fg(m+1) = g(\mathbf{r}_\sigma fg(m))m$.

Let $\mathbb{C}$ be such that $\mathcal{T}(\mathbb{C}) = \mathcal{T}(\lambda\mathsf{T})$ and define $\mu : |\mathbb{C}| \to |\mathfrak{M}(\omega)|$ by

(i) $\mu(\mathbf{c}_m) = m$

(ii) $\mu(\mathbf{S}^+) = S^+$

(iii) $\mu(\mathbf{R}_\sigma) = \mathbf{r}_\sigma$.

Then $(\mathfrak{M}(\omega), [\![\;]\!]^\mu)$ is a $\lambda\mathsf{T}$-model, by 2.2.15. □

In the following we will look at algebras and models for systems Σ_1 and Σ_2 for which we have $\Sigma_1 \trianglelefteq \Sigma_2$.

Let Σ_1 and Σ_2 be systems and let $\Xi : \mathcal{T}(\Sigma_1) \to \mathcal{T}(\Sigma_2)$ be a transformation. Let $(\mathfrak{M}, [\![\;]\!]^2)$ be a quasi-Σ_2-interpretation.

2.2.20. DEFINITION. Let $[\![\;]\!]^1 : \mathcal{T}(\Sigma_1) \times Val(\mathfrak{M}) \to |\mathfrak{M}|$ be defined by

$$[\![M]\!]^1_\rho = [\![\Xi(M)]\!]^2_\rho. \;\square$$

2.2.21. LEMMA. *$(\mathfrak{M}, [\![\;]\!]^1)$ is a quasi-Σ_1-structure.*

PROOF. It is easy to show that $[\![\;]\!]^1$ is a quasi-Σ_1-interpretation in $\mathfrak{M}$. □

2.2.22. PROPOSITION. (i) $\mathfrak{M}, [\![\;]\!]^1 \models M = N \Longleftrightarrow \mathfrak{M}, [\![\;]\!]^2 \models \Xi(M) = \Xi(N)$.

(ii) $\mathfrak{M}, [\![\;]\!]^1 \models_q M = N \Longleftrightarrow \mathfrak{M}, [\![\;]\!]^2 \models_q \Xi(M) = \Xi(N)$.

PROOF. Simple. □

2.2.23. THEOREM. *Let $\Xi : \Sigma_1 \to \Sigma_2$ be a translation.*
(i) *If $(\mathfrak{M}, [\![\,]\!]^2)$ is a Σ_2-model, then $(\mathfrak{M}, [\![\,]\!]^1)$ is a Σ_1-model.*
(ii) *If $(\mathfrak{M}, [\![\,]\!]^2)$ is a Σ_2-algebra, then $(\mathfrak{M}, [\![\,]\!]^1)$ is a Σ_1-algebra.*
(iii) *If $(\mathfrak{M}, [\![\,]\!]^2)$ is a quasi-Σ_2-model, then $(\mathfrak{M}, [\![\,]\!]^1)$ is a quasi-Σ_1-model.*
(iv) *If $(\mathfrak{M}, [\![\,]\!]^2)$ is a quasi-Σ_2-algebra, then $(\mathfrak{M}, [\![\,]\!]^1)$ is a quasi-Σ_1-algebra.*

PROOF. By 2.2.22. □

2.3. λ-Structures.

A λ-structure is a quasi-extensional type structure in which every term in $\mathbf{CL}(\mathbb{C}^*)$ can be interpreted in such a way that w-convertible terms have extensionally equal interpretations. We will show that the λ-structures are exactly the quasi-extensional type structures that carry quasi-$\mathcal{T}(\mathbb{C})$-interpretations for every $\mathbb{C}$.

2.3.1. DEFINITION. (i) A quasi-k-family in $\mathfrak{M}$ is a family $(\mathbf{k}_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}$ of objects in $\mathfrak{M}$ such that for all $\sigma, \tau \in \mathbb{T}$ $\mathbf{k}_{\sigma,\tau}$ is of type $\sigma \to \tau \to \sigma$ and for all $a \in \mathfrak{M}_\sigma$ and $b \in \mathfrak{M}_\tau$

$$\mathbf{k}_{\sigma,\tau} ab \approx a.$$

(ii) A quasi-s-family in $\mathfrak{M}$ is a family $(\mathbf{s}_{\sigma,\tau,\upsilon})_{\sigma,\tau,\upsilon\in\mathbb{T}}$ of objects in $\mathfrak{M}$ such that for all $\sigma, \tau, \upsilon \in \mathbb{T}$ $\mathbf{s}_{\sigma,\tau,\upsilon}$ is of type $(\sigma \to \tau \to \upsilon) \to (\sigma \to \tau) \to \sigma \to \upsilon$ and for all $a \in \mathfrak{M}_{\sigma\to\tau\to\upsilon}$, $b \in \mathfrak{M}_{\sigma\to\tau}$ and $c \in \mathfrak{M}_\sigma$

$$\mathbf{s}_{\sigma,\tau,\upsilon} abc \approx ac(bc).$$

(iii) $\mathfrak{M}$ is quasi-combinatory if $\mathfrak{M}$ has a quasi-k-family and a quasi-s-family. □

2.3.2. LEMMA. *If $(\mathfrak{M}, [\![\,]\!])$ is a quasi-$\mathcal{T}(\mathbb{C})$-structure, then $\mathfrak{M}$ is quasi-combinatory.*

PROOF. (i) For all $\sigma, \tau \in \mathbb{T}$ define $\mathbf{k}_{\sigma,\tau} = [\![\mathbf{K}_{\sigma,\tau}]\!]$. Then $(\mathbf{k}_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}$ is a quasi-k-family.

(ii) For all $\sigma, \tau, \upsilon \in \mathbb{T}$ define $\mathbf{s}_{\sigma,\tau,\upsilon} = [\![\mathbf{S}_{\sigma,\tau,\upsilon}]\!]$. Then $(\mathbf{s}_{\sigma,\tau,\upsilon})_{\sigma,\tau,\upsilon\in\mathbb{T}}$ is a quasi-s-family. □

2.3.3. DEFINITION. $\mathfrak{M}$ is a λ-structure if $\mathfrak{M}$ is quasi-extensional and quasi-combinatory. □

We are going to show that every λ-structure carries quasi-$\mathcal{T}(\mathbb{C})$-interpretations for every $\mathbb{C}$. For this we will need $\mathbf{CL}(\mathbb{C}^*)$-interpretations. These will be defined first.

2.3.4. DEFINITION. (i) For all $\rho_1, \rho_2 \in Val(\mathfrak{M})$ and $P \in \mathbf{CL}(\mathbb{C})$

$$\rho_1 =_P \rho_2 \Longleftrightarrow \forall x \in FV(P)[\rho_1(x) = \rho_2(x)].$$

(ii) A $\mathbf{CL}(\mathbb{C})$-interpretation in $\mathfrak{M}$ is a mapping

$$(\ ) : \mathbf{CL}(\mathbb{C}) \times Val(\mathfrak{M}) \to |\mathfrak{M}|$$

$$(P, \rho) \mapsto (P)_\rho$$

such that

(1) $P \in \mathbf{CL}_\sigma(\mathbb{C}) \Rightarrow (P)_\rho \in \mathfrak{M}_\sigma$.

(2) $(x)_\rho = \rho(x)$.

(3) $(PQ)_\rho = (P)_\rho (Q)_\rho$.

(4) $\rho_1 =_P \rho_2 \Rightarrow (P)_{\rho_1} = (P)_{\rho_2}$. $\square$

Let $\mathfrak{M}$ be a λ-structure. Let $(\mathbf{k}_{\sigma,\tau})_{\sigma,\tau \in \mathbb{T}}$ be a quasi-k-family in $\mathfrak{M}$ and let $(\mathbf{s}_{\sigma,\tau,\upsilon})_{\sigma,\tau,\upsilon \in \mathbb{T}}$ be a quasi-s-family in $\mathfrak{M}$.

Let $\mathbb{C}$ be a constant family. Let $\mu : |\mathbb{C}| \to |\mathfrak{M}|$ be type-preserving. In the following we will construct a quasi-$\mathcal{T}(\mathbb{C})$-interpretation $[\![\ ]\!]$ in $\mathfrak{M}$ such that $[\![\mathbf{C}]\!] = \mu(\mathbf{C})$ for every $\mathbf{C} \in |\mathbb{C}|$.

2.3.5. DEFINITION. Let $(\ ) : \mathbf{CL}(\mathbb{C}^*) \times Val(\mathfrak{M}) \to |\mathfrak{M}|$ be defined by induction on the structure of the terms in $\mathbf{CL}(\mathbb{C}^*)$ as follows.

(i) $(\mathsf{K}_{\sigma,\tau})_\rho = \mathbf{k}_{\sigma,\tau}$.
(ii) $(\mathsf{S}_{\sigma,\tau,\upsilon})_\rho = \mathbf{s}_{\sigma,\tau,\upsilon}$.
(iii) $(\mathbf{C})_\rho = \mu(\mathbf{C})$.
(iv) $(x)_\rho = \rho(x)$.
(v) $(PQ)_\rho = (P)_\rho (Q)_\rho$. $\square$

By induction on the structure of P it can be shown that

- $P \in \mathbf{CL}_\sigma(\mathbb{C}^*) \Rightarrow (P)_\rho \in \mathfrak{M}_\sigma$
- $\rho_1 =_P \rho_2 \Rightarrow (P)_{\rho_1} = (P)_{\rho_2}$.

Therefore $(\ )$ is a $\mathbf{CL}(\mathbb{C}^*)$-interpretation in $\mathfrak{M}$.

2.3.6. PROPOSITION. $(\lambda^* x.P)_\rho \approx (P)_{\rho(x:=a)}$.

PROOF. By induction on the structure of P.

(i) Say $x \in Var_\sigma$. Then

$$\begin{aligned}(\lambda^* x.x)_\rho a &= (\mathbf{I}_\sigma)_\rho a\\ &= \mathbf{s}_{\sigma,\sigma\to\sigma,\sigma}\mathbf{k}_{\sigma,\sigma\to\sigma}\mathbf{k}_{\sigma,\sigma} a\\ &\approx \mathbf{k}_{\sigma,\sigma\to\sigma} a(\mathbf{k}_{\sigma,\sigma} a)\\ &\approx a\\ &= (x)_{\rho(x:=a)}.\end{aligned}$$

(ii) If $x \notin FV(P)$, then

$$\begin{aligned}(\lambda^* x.P)_\rho a &= (\mathsf{K}_{\tau,\sigma} P)_\rho a\\ &= \mathbf{k}_{\tau,\sigma}(P)_\rho a\\ &\approx (P)_\rho\\ &= (P)_{\rho(x:=a)}.\end{aligned}$$

(iii) If $x \in FV(PQ)$, then

$$\begin{aligned}(\lambda^* x.PQ)_\rho a &= (\mathsf{S}_{\sigma,\tau,\upsilon}(\lambda^* x.P)(\lambda^* x.Q))_\rho a\\ &= \mathbf{s}_{\sigma,\tau,\upsilon}(\lambda^* x.P)_\rho(\lambda^* x.Q)_\rho a\\ &\approx (\lambda^* x.P)_\rho a((\lambda^* x.Q)_\rho a)\\ &\approx (P)_{\rho(x:=a)}(Q)_{\rho(x:=a)}\\ &= (PQ)_{\rho(x:=a)}. \ \square\end{aligned}$$

2.3.7. Definition. Let $[\![\ ]\!] : \mathcal{T}(\mathbb{C}) \times Val(\mathfrak{M}) \to |\mathfrak{M}|$ be defined by

$$[\![M]\!]_\rho = (M_{CL})_\rho. \ \square$$

2.3.8. Lemma. *$[\![\,]\!]$ is a quasi-$\mathcal{T}(\mathbb{C})$-interpretation in $\mathfrak{M}$ with $[\![\mathbf{C}]\!] = \mu(\mathbf{C})$ for every $\mathbf{C} \in |\mathbb{C}|$.*

Proof. We only verify the condition $[\![\lambda x.M]\!]_\rho a \approx [\![M]\!]_{\rho(x:=a)}$. This is done as follows.

$$\begin{aligned}[\![\lambda x.M]\!]_\rho a &= ((\lambda x.M)_{CL})_\rho a\\ &= (\lambda^* x.M_{CL})_\rho a\\ &\approx (M_{CL})_{\rho(x:=a)}\\ &= [\![M]\!]_{\rho(x:=a)}. \ \square\end{aligned}$$

2.3.9. Theorem. *Let $\mathfrak{M}$ be a λ-structure. Let $\mathbb{C}$ be a constant family. Let $\mu : |\mathbb{C}| \to |\mathfrak{M}|$ be type-preserving. Then there exists a quasi-$\mathcal{T}(\mathbb{C})$-interpretation $[\![\ ]\!]$ in $\mathfrak{M}$ such that $[\![\mathbf{C}]\!] = \mu(\mathbf{C})$ for every $\mathbf{C} \in |\mathbb{C}|$.*

PROOF. By 2.3.5-2.3.8. □

2.3.10. THEOREM. *Let $\mathfrak{M}$ be quasi-extensional. Let $\mathbb{C}$ be a constant family. Then $\mathfrak{M}$ is quasi-combinatory if and only if there exists a quasi-$\mathcal{T}(\mathbb{C})$-interpretation in $\mathfrak{M}$.*

PROOF. By 2.3.2 and 2.3.9. □

Next we will take a short look at combinatory type structures.

2.3.11. DEFINITION. (i) A k-family in $\mathfrak{M}$ is a family $(\mathbf{k}_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}$ of objects in $\mathfrak{M}$ such that for all $\sigma,\tau \in \mathbb{T}$ $\mathbf{k}_{\sigma,\tau}$ is of type $\sigma\rightarrow\tau\rightarrow\sigma$ and for all $a \in \mathfrak{M}_\sigma$ and $b \in \mathfrak{M}\tau$

$$\mathbf{k}_{\sigma,\tau}ab = a.$$

(ii) An s-family in $\mathfrak{M}$ is a family $(\mathbf{s}_{\sigma,\tau,\upsilon})_{\sigma,\tau,\upsilon\in\mathbb{T}}$ of objects in $\mathfrak{M}$ such that for all $\sigma,\tau,\upsilon \in \mathbb{T}$ $\mathbf{s}_{\sigma,\tau,\upsilon}$ is of type $(\sigma\rightarrow\tau\rightarrow\upsilon)\rightarrow(\sigma\rightarrow\tau)\rightarrow\sigma\rightarrow\upsilon$ such that for all $a \in \mathfrak{M}_{\sigma\rightarrow\tau\rightarrow\upsilon}$, $b \in \mathfrak{M}_{\sigma\rightarrow\tau}$ and $c \in \mathfrak{M}_\sigma$

$$\mathbf{s}_{\sigma,\tau,\upsilon}abc = ac(bc).$$

(iii) $\mathfrak{M}$ is combinatory if $\mathfrak{M}$ has a k-family and an s-family. □

2.3.12. THEOREM. *Let $\mathfrak{M}$ be combinatory. Let $\mathbb{C}$ be a constant family. Let $\mu : |\mathbb{C}| \rightarrow |\mathfrak{M}|$ be type-preserving. Then there exists a $\mathcal{T}(\mathbb{C})$-interpretation $[\![\ ]\!]$ in $\mathfrak{M}$ such that $[\![\mathbf{C}]\!] = \mu(\mathbf{C})$ for every $\mathbf{C} \in |\mathbb{C}|$.*

PROOF. Similar to the proof of 2.3.9. Only this time we do not need to assume that $\mathfrak{M}$ is quasi-extensional or extensional. □

2.3.13. THEOREM. *Let $\mathbb{C}$ be a constant family. Then $\mathfrak{M}$ is combinatory if and only if there exists a $\mathcal{T}(\mathbb{C})$-interpretation in $\mathfrak{M}$.* □

Theorem 2.3.9 is an existence theorem. For quasi-interpretations in λ-structures we also have a unicity theorem.

2.3.14. DEFINITION. Let $[\![\ ]\!]^1$ and $[\![\ ]\!]^2$ be two quasi-$\mathcal{T}(\mathbb{C})$-interpretations in $\mathfrak{M}$. Then $[\![\ ]\!]^1$ and $[\![\ ]\!]^2$ are equivalent (notation $[\![\ ]\!]^1 \approx [\![\ ]\!]^2$) if for all $M \in \mathcal{T}(\mathbb{C})$ and all $\rho \in Val(\mathfrak{M})$ we have $[\![M]\!]^1_\rho \approx [\![M]\!]^2_\rho$. □

2.3.15. THEOREM. *Let $\mathfrak{M}$ be quasi-extensional. Let $[\![\ ]\!]^1$ and $[\![\ ]\!]^2$ be two quasi-$\mathcal{T}(\mathbb{C})$-interpretations in $\mathfrak{M}$ such that $[\![\mathbf{C}]\!]^1 \approx [\![\mathbf{C}]\!]^2$ for all $\mathbf{C} \in |\mathbb{C}|$. Then $[\![]\!]^1 \approx [\![]\!]^2$.*

PROOF. By induction on the structure of M it will be shown that for all $M \in \mathcal{T}(\mathbb{C})$ and all $\rho \in Val(\mathfrak{M})$

$$[\![M]\!]^1_\rho \approx [\![M]\!]^2_\rho.$$

(i) $[\![\mathbf{C}]\!]^1_\rho = [\![\mathbf{C}]\!]^1 \approx [\![\mathbf{C}]\!]^2 = [\![\mathbf{C}]\!]^2_\rho$.
(ii) $[\![x]\!]^1_\rho = \rho(x) = [\![x]\!]^2_\rho$.
(iii) By induction $[\![M]\!]^1_\rho \approx [\![M]\!]^2_\rho$ and $[\![N]\!]^1_\rho \approx [\![N]\!]^2_\rho$. Then, by quasi-extensionality, we have

$$\begin{array}{rcl} [\![MN]\!]^1_\rho & = & [\![M]\!]^1_\rho [\![N]\!]^1_\rho \\ & \approx & [\![M]\!]^2_\rho [\![N]\!]^2_\rho \\ & = & [\![MN]\!]^2_\rho. \end{array}$$

(iv) Let $\sigma \in \mathbb{T}$, $x \in Var_\sigma$. Then for all $a \in \mathfrak{M}_\sigma$

$$\begin{array}{rcl} [\![\lambda x.M]\!]^1_\rho a & \approx & [\![M]\!]^1_{\rho(x:=a)} \\ & \approx & [\![M]\!]^2_{\rho(x:=a)} \\ & \approx & [\![\lambda x.M]\!]^2_\rho a \end{array}$$

because by induction $[\![M]\!]^1_{\rho(x:=a)} \approx [\![M]\!]^2_{\rho(x:=a)}$. Therefore $[\![\lambda x.M]\!]^1_\rho \approx [\![\lambda x.M]\!]^2_\rho$. □

2.3.16. Corollary. *Let $\mathfrak{M}$ be extensional. Let $[\![]\!]^1$ and $[\![]\!]^2$ be two $\mathcal{T}(\mathbb{C})$-interpretations in $\mathfrak{M}$ such that $[\![\mathbf{C}]\!]^1 = [\![\mathbf{C}]\!]^2$ for every $\mathbf{C} \in |\mathbb{C}|$. Then $[\![\ ]\!]^1 = [\![\ ]\!]^2$.* □

Next we will look at sets of objects in type structures.

2.3.17. Definition. Let $\mathbb{O} \subseteq |\mathfrak{M}|$.
(i) For every $\sigma \in \mathbb{T}$ define $\mathbb{O}_\sigma = \mathbb{O} \cap \mathfrak{M}_\sigma$.
(ii) The set $\mathbb{O}$ is called a λ-set if

(1) for all $\sigma \in \mathbb{T}$ and $a, b \in \mathfrak{M}_\sigma$ we have $a \in \mathbb{O} \;\&\; a \approx b \Rightarrow b \in \mathbb{O}$
(2) for all $\sigma, \tau \in \mathbb{T}$, $a \in \mathfrak{M}_{\sigma\to\tau}$, $b \in \mathfrak{M}_\sigma$ we have $a, b \in \mathbb{O} \Rightarrow ab \in \mathbb{O}$
(3) there exists a quasi-k-family $(\mathbf{k}_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}$ in $\mathfrak{M}$ such that $\mathbf{k}_{\sigma,\tau} \in \mathbb{O}$ for all $\sigma, \tau \in \mathbb{T}$.
(4) there exists a quasi-s-family $(\mathbf{s}_{\sigma,\tau,\upsilon})_{\sigma,\tau,\upsilon\in\mathbb{T}}$ in $\mathfrak{M}$ such that $\mathbf{s}_{\sigma,\tau,\upsilon} \in \mathbb{O}$ for all $\sigma, \tau, \upsilon \in \mathbb{T}$. □

If $\mathfrak{M}$ is a λ-structure, and $[\![\ ]\!]$ is a quasi-$\mathcal{T}(\mathbb{C})$-interpretation in $\mathfrak{M}$, then the closed terms in $\mathcal{T}(\mathbb{C})$ can be seen as definitions for objects in $\mathfrak{M}$. This gives a notion of definability in $(\mathfrak{M}, [\![\ ]\!])$.

2.3.18. Definition. Let $\mathfrak{M}$ be a λ-structure, and let $[\![]\!]$ be a quasi-$\mathcal{T}(\mathbb{C})$-interpretation in $\mathfrak{M}$.
(i) An object a of $\mathfrak{M}$ is definable if there exists $M \in \mathcal{T}(\mathbb{C})^0$ such that $[\![M]\!] \approx a$. If $M \in \mathcal{T}(\mathbb{C})^0$ is such that $[\![M]\!] \approx a$, then M is said to define a.
(ii) If $\sigma \in \mathbb{T}$, then $\mathfrak{D}_\sigma(\mathfrak{M}, [\![\ ]\!])$ is the set of all definable type-σ objects of $(\mathfrak{M}, [\![\ ]\!])$.

(iii) $\mathfrak{D}(\mathfrak{M}, [\![\;]\!])$ is the set of all definable objects of $(\mathfrak{M}, [\![\;]\!])$. □

2.3.19. THEOREM. *Let $\mathfrak{M}$ be a λ-structure, and let $[\![\,]\!]$ be a quasi-$\mathcal{T}(\mathbb{C})$-interpretation in $\mathfrak{M}$. Then $\mathfrak{D}(\mathfrak{M}, [\![\;]\!])$ is a λ-set.*

PROOF. (i) Let $a \in \mathfrak{D}(\mathfrak{M}, [\![\;]\!])$ and $a \approx b$. Then $a \approx [\![M]\!]$ for some $M \in \mathcal{T}(\mathbb{C})^0$. Then also $b \approx [\![M]\!]$ and so $b \in \mathfrak{D}(\mathfrak{M}, [\![\;]\!])$.

(ii) Let $\sigma, \tau \in \mathbb{T}$, $a \in \mathfrak{D}_{\sigma\to\tau}(\mathfrak{M}, [\![\;]\!])$, $b \in \mathfrak{D}_\sigma(\mathfrak{M}, [\![\;]\!])$. Then there exist $M \in \mathcal{T}_{\sigma\to\tau}(\mathbb{C})^0$ and $N \in \mathcal{T}_\sigma(\mathbb{C})^0$ such that $a \approx [\![M]\!]$ and $b \approx [\![N]\!]$. Then, by quasi-extensionality $ab \approx [\![M]\!][\![N]\!] = [\![MN]\!]$. So $ab \in \mathfrak{D}_\tau(\mathfrak{M}, [\![\;]\!])$.

(iii) For all $\sigma, \tau \in \mathbb{T}$ define $\mathbf{k}_{\sigma,\tau} = [\![\mathbf{K}_{\sigma,\tau}]\!]$. Then $(\mathbf{k}_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}$ is a quasi-k-family in $\mathfrak{M}$ with $\mathbf{k}_{\sigma,\tau} \in \mathfrak{D}(\mathfrak{M}, [\![\;]\!])$ for all $\sigma, \tau \in \mathbb{T}$.

(iv) For all $\sigma, \tau, \upsilon \in \mathbb{T}$ define $\mathbf{s}_{\sigma,\tau,\upsilon} = [\![\mathbf{S}_{\sigma,\tau,\upsilon}]\!]$. Then $(\mathbf{s}_{\sigma,\tau,\upsilon})_{\sigma,\tau,\upsilon\in\mathbb{T}}$ is a quasi-s-family in $\mathfrak{M}$ with $\mathbf{s}_{\sigma,\tau,\upsilon} \in \mathfrak{D}(\mathfrak{M}, [\![\;]\!])$ for all $\sigma, \tau, \upsilon \in \mathbb{T}$. □

2.3.20. THEOREM. *Let $\mathfrak{M}$ be a λ-structure. Let $\mathbb{O} \subseteq |\mathfrak{M}|$ be a λ-set. Let $[\![\;]\!]$ be a quasi-$\mathcal{T}(\mathbb{C})$-interpretation in $\mathfrak{M}$ such that $[\![\mathbf{C}]\!] \in \mathbb{O}$ for all $\mathbf{C} \in |\mathbb{C}|$. Then for every $\sigma \in \mathbb{T}$*

$$\mathfrak{D}_\sigma(\mathfrak{M}, [\![\;]\!]) \subseteq \mathbb{O}_\sigma.$$

PROOF. Let $(\mathbf{k}_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}$ be a quasi-k-family in $\mathfrak{M}$ such that $\mathbf{k}_{\sigma,\tau} \in \mathbb{O}$ for all $\sigma, \tau \in \mathbb{T}$ and let $(\mathbf{s}_{\sigma,\tau,\upsilon})_{\sigma,\tau,\upsilon\in\mathbb{T}}$ be a quasi-s-family in $\mathfrak{M}$ such that $\mathbf{s}_{\sigma,\tau,\upsilon} \in \mathbb{O}$ for all $\sigma, \tau, \upsilon \in \mathbb{T}$.

Let $(\;)$ be the $\mathbf{CL}(\mathbb{C}^*)$-interpretation in $\mathfrak{M}$ such that

(i) $(\mathsf{K}_{\sigma,\tau})_\rho = \mathbf{k}_{\sigma,\tau}$

(ii) $(\mathsf{S}_{\sigma,\tau,\upsilon})_\rho = \mathbf{s}_{\sigma,\tau,\upsilon}$

(iii) $(\mathbf{C})_\rho = [\![\mathbf{C}]\!]$.

and let $[\![\;]\!]^1$ be the quasi-$\mathcal{T}(\mathbb{C})$-interpretation in $\mathfrak{M}$ such that

$$[\![M]\!]^1_\rho = (M_{CL})_\rho.$$

Using that $\mathbb{O}$ is closed under application if follows by induction on the structure of P that $(P) \in \mathbb{O}$ for all $P \in \mathbf{CL}_\sigma(\mathbb{C}^*)^0$. Therefore we have that $[\![M]\!]^1 \in \mathbb{O}$ for all $M \in \mathcal{T}(\mathbb{C})^0$. Because $\mathfrak{M}$ is a λ-structure and $[\![\mathbf{C}]\!] = [\![\mathbf{C}]\!]^1$ for all $\mathbf{C} \in |\mathbb{C}|$ we have that $[\![\;]\!] \approx [\![\;]\!]^1$. Therefore $[\![M]\!] \in \mathbb{O}$ for all $M \in \mathcal{T}(\mathbb{C})^0$, because $\mathbb{O}$ is closed under extensional equality.

Now let $\sigma \in \mathbb{T}$ and $a \in \mathfrak{D}_\sigma(\mathfrak{M}, [\![\;]\!])$. Then there exists $M \in \mathcal{T}_\sigma(\mathbb{C})^0$ with $a \approx [\![M]\!]$. Then $[\![M]\!] \in \mathbb{O}_\sigma$ and so $a \in \mathbb{O}_\sigma$, again because $\mathbb{O}$ is closed under extensional equality. □

2.4. The Zucker collapse.

The Zucker collapse is an operation on type structures. If a type structure $\mathfrak{M}$ has the so-called Zucker Property, then the Zucker collapse can be applied to $\mathfrak{M}$ yielding a quasi-extensional substructure $\mathfrak{M}^Z$ of $\mathfrak{M}$. The type structure $\mathfrak{M}^Z$ is called the Zucker collapse of $\mathfrak{M}$. The Zucker collapse is due to [Zucker 1971].

First we define substructures. The type structure $\mathfrak{M}^1$ is a substructure of $\mathfrak{M}^2$ (notation $\mathfrak{M}^1 \subseteq \mathfrak{M}^2$) if for every $\sigma \in \mathbb{T}$ we have $\mathfrak{M}^1_\sigma \subseteq \mathfrak{M}^2_\sigma$ and for all $\sigma, \tau \in \mathbb{T}$, $a \in \mathfrak{M}^1_{\sigma\to\tau}$, $b \in \mathfrak{M}^1_\sigma$ we have $App^1_{\sigma,\tau}(a,b) = App^2_{\sigma,\tau}(a,b)$. If $\mathfrak{M}^1 \subseteq \mathfrak{M}^2$, then any two objects of $\mathfrak{M}^1$ that are extensionally equal in $\mathfrak{M}^2$ are also extensionally equal in $\mathfrak{M}^1$.

In the following we will define the Zucker Property for type structures, and for every type structure $\mathfrak{M}$ having the Zucker Property we will define the Zucker collapse $\mathfrak{M}^Z$ of $\mathfrak{M}$.

2.4.1. Definition. For every $\sigma \in \mathbb{T}$ define a binary relation $\sim_\sigma$ on $\mathfrak{M}_\sigma$ by induction on the structure of σ as follows.

(i) For all $a_1, a_2 \in \mathfrak{M}_0$

$$a_1 \sim_0 a_2 \iff a_1 = a_2.$$

(ii) For all $a_1, a_2 \in \mathfrak{M}_{\sigma\to\tau}$

$$a_1 \sim_{\sigma\to\tau} a_2 \iff \forall b_1, b_2 \in \mathfrak{M}_\sigma [\, b_1 \sim_\sigma b_2 \Rightarrow a_1 b_1 \sim_\tau a_2 b_2 \,]. \ \square$$

Instead of $a_1 \sim_\sigma a_2$ we usually write $a_1 \sim a_2$. If $\vec{a} = (a_1, \ldots, a_k)$ and $\vec{b} = (b_1, \ldots, b_k)$, then

$$\vec{a} \sim \vec{b} \iff a_i \sim b_i \ \ (i = 1, \ldots, k).$$

By induction on the structure of σ it can be shown that for all $\sigma \in \mathbb{T}$ and $a, b \in \mathfrak{M}_\sigma$

$$a \sim b \iff \forall \vec{c}, \vec{d} \in (\mathfrak{M})_\sigma [\, \vec{c} \sim \vec{d} \Rightarrow a\vec{c} = b\vec{d}\,].$$

2.4.2. Lemma. (i) *$a_1 \sim_{\sigma\to\tau} a_2$ & $b_1 \sim_\sigma b_2 \Rightarrow a_1 b_1 \sim_\tau a_2 b_2$.*

(ii) *$\sim_\sigma$ is a partial equivalence relation on $\mathfrak{M}_\sigma$.*

Proof. (i) Immediately from definition 2.4.1.

(ii) by induction on the structure of σ.

(1) The relation $\sim_0$ coincides with equality on $\mathfrak{M}_0$ and is therefore symmetric and transitive.

(2) Let $\sigma, \tau \in \mathbb{T}$ and suppose that $\sim_\sigma$ and $\sim_\tau$ are partial equivalence relations.

- Let $a_1 \sim_{\sigma\to\tau} a_2$. Let $b_1, b_2 \in \mathfrak{M}_\sigma$ be such that $b_1 \sim_\sigma b_2$. By induction $b_2 \sim_\sigma b_1$, so $a_1 b_2 \sim_\tau a_2 b_1$, and again by induction $a_2 b_1 \sim_\tau a_1 b_2$. Therefore $a_2 \sim_{\sigma\to\tau} a_1$. So $\sim_{\sigma\to\tau}$ is symmetric.
- Let $a_1 \sim_{\sigma\to\tau} a_2$ and $a_2 \sim_{\sigma\to\tau} a_3$. Let $b_1 \sim_\sigma b_2$. Then $b_1 \sim_\sigma b_1$ (the elements in the field of a partial equivalence relation are reflexive), so $a_1 b_1 \sim_\tau a_2 b_1$ and $a_2 b_2 \sim_\tau a_3 b_2$. By induction $a_1 b_1 \sim_\tau a_3 b_2$. Therefore $a_1 \sim_{\sigma\to\tau} a_3$. So $\sim_{\sigma\to\tau}$ is transitive. $\Box$

2.4.3. DEFINITION. The type structure $\mathfrak{M}$ has the Zucker Property if for every $\sigma \in \mathbb{T}$ there exists $a \in \mathfrak{M}_\sigma$ with $a \sim a$. $\Box$

2.4.4. DEFINITION. Let $\mathfrak{M}$ have the Zucker Property.
(i) For every $\sigma \in \mathbb{T}$ define $\mathfrak{M}^Z_\sigma = \{a \in \mathfrak{M}_\sigma \mid a \sim a\}$.
(ii) The Zucker collapse of $\mathfrak{M}$ is the type structure

$$\mathfrak{M}^Z = ((\mathfrak{M}^Z_\sigma)_{\sigma\in\mathbb{T}}, (App^Z_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}})$$

where $App^Z_{\sigma,\tau} : \mathfrak{M}^Z_{\sigma\to\tau} \times \mathfrak{M}^Z_\sigma \to \mathfrak{M}^Z_\tau$ is defined by $App^Z_{\sigma,\tau}(a,b) = App_{\sigma,\tau}(a,b)$. $\Box$

Because $\mathfrak{M}$ has the Zucker Property $\mathfrak{M}^Z$ is indeed a type structure. Note that $\mathfrak{M}^Z \subseteq \mathfrak{M}$. Next it will be shown that $\mathfrak{M}^Z$ is quasi-extensional.

2.4.5. LEMMA. *Let $\approx^Z$ denote extensional equality in $\mathfrak{M}^Z$. Then for all $\sigma \in \mathbb{T}$*
(i) *a is $\mathfrak{M}^Z$-extensional for every $a \in \mathfrak{M}^Z_\sigma$*
(ii) $\forall a_1, a_2 \in \mathfrak{M}^Z_\sigma [\, a_1 \approx^Z a_2 \iff a_1 \sim a_2 \,]$.

PROOF. By induction on the structure of σ. For $\sigma = 0$ the proof is simple. Let $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0 \neq 0$ and suppose (i) and (ii) hold for $\sigma_1, \ldots, \sigma_k$.

(i) Let $a \in \mathfrak{M}^Z_\sigma$. Then $a \in \mathfrak{M}_\sigma$ and $a \sim a$. Let $\vec{b}, \vec{c} \in (\mathfrak{M}^Z)_\sigma$ be such that $\vec{b} \approx^Z \vec{c}$. Then $\vec{b} \sim \vec{c}$ by induction, so $a\vec{b} = a\vec{c}$. Therefore a is $\mathfrak{M}^Z$-extensional.

(ii) Let $a_1, a_2 \in \mathfrak{M}^Z_\sigma$.

- Suppose $a_1 \approx^Z a_2$. Let $\vec{b}, \vec{c} \in (\mathfrak{M})_\sigma$ with $\vec{b} \sim \vec{c}$. Then $\vec{b}, \vec{c} \in (\mathfrak{M}^Z)_\sigma$ and so $\vec{b} \approx^Z \vec{c}$ by induction. By (i) a_1 is $\mathfrak{M}^Z$-extensional so we have $a_1\vec{b} = a_1\vec{c}$. Also $a_1\vec{c} = a_2\vec{c}$ because $a_1 \approx^Z a_2$. Therefore $a_1\vec{b} = a_2\vec{c}$. So $a_1 \sim a_2$.
- Now suppose $a_1 \sim a_2$. Let $\vec{b} \in (\mathfrak{M}^Z)_\sigma$. Then $\vec{b} \sim \vec{b}$ and so $a_1\vec{b} = a_2\vec{b}$. Therefore $a_1 \approx^Z a_2$. $\Box$

2.4.6. PROPOSITION. *Extensional equality in $\mathfrak{M}^Z$ coincides with $\sim$.* $\Box$

2.4.7. THEOREM. *$\mathfrak{M}^Z$ is quasi-extensional.* $\Box$

Because $\mathfrak{M}^Z \subseteq \mathfrak{M}$ we have that for all $\sigma \in \mathbb{T}$ and $a, b \in \mathfrak{M}^Z_\sigma$ $a \approx b \Rightarrow a \sim b$.

In the following it will be shown that if $\mathfrak{M}$ is a quasi-extensional type structure with the Zucker Property, then $\mathfrak{M}^Z = \mathfrak{M}$.

2.4.8. LEMMA. *$\mathfrak{M}$ is quasi-extensional if and only if for every $\sigma \in \mathbb{T}$ and all $a_1, a_2 \in \mathfrak{M}_\sigma$*

$$a_1 \approx a_2 \Longleftrightarrow a_1 \sim a_2.$$

PROOF. (i) Let $\mathfrak{M}$ be quasi-extensional. Use induction on the structure of σ.

(1) For all $a_1, a_2 \in \mathfrak{M}_0$: $a_1 \approx a_2 \Leftrightarrow a_1 = a_2 \Leftrightarrow a_1 \sim a_2$.

(2) Let $a_1, a_2 \in \mathfrak{M}_{\sigma \to \tau}$.

- Suppose $a_1 \approx a_2$. Let $b_1, b_2 \in \mathfrak{M}_\sigma$ with $b_1 \sim b_2$. Then $b_1 \approx b_2$ by induction. So $a_1 b_1 \approx a_2 b_2$ by quasi-extensionality, and again by induction $a_1 b_1 \sim a_2 b_2$. Therefore $a_1 \sim a_2$.
- Now suppose $a_1 \sim a_2$. Let $b \in \mathfrak{M}_\sigma$. Then $b \approx b$, so $b \sim b$ by induction. Therefore $a_1 b \sim a_2 b$ and again by induction $a_1 b \approx a_2 b$. So $a_1 \approx a_2$.

(ii) Let $\forall \sigma \in \mathbb{T} \forall a_1, a_2 \in \mathfrak{M}_\sigma [\, a_1 \approx a_2 \Longleftrightarrow a_1 \sim a_2 \,]$. Let $\sigma, \tau \in \mathbb{T}$, $a_1, a_2 \in \mathfrak{M}_{\sigma \to \tau}$, $b_1, b_2 \in \mathfrak{M}_\sigma$ be such that $a_1 \approx a_2$ and $b_1 \approx b_2$. Then $a_1 \sim a_2$ and $b_1 \sim b_2$. So $a_1 b_1 \sim a_2 b_2$ and so $a_1 b_1 \approx a_2 b_2$. Therefore $\mathfrak{M}$ is quasi-extensional. □

2.4.9. LEMMA. *The following are equivalent:*

(i) $\forall \sigma \in \mathbb{T} \forall a_1, a_2 \in \mathfrak{M}_\sigma [\, a_1 \approx a_2 \Leftrightarrow a_1 \sim a_2 \,]$.

(ii) $\forall \sigma \in \mathbb{T} \forall a \in \mathfrak{M}_\sigma [\, a \sim a \,]$.

PROOF. (i)⇒(ii) Suppose (i). Let $\sigma \in \mathbb{T}$, $a \in \mathfrak{M}_\sigma$. Then $a \approx a$ and so $a \sim a$ by (i).

(ii)⇒(i) Now suppose (ii). We show (i) by induction on the structure of σ. For $\sigma = 0$ the proof is simple. Then let $\sigma, \tau \in \mathbb{T}$. Let $a_1, a_2 \in \mathfrak{M}_{\sigma \to \tau}$.

- Suppose $a_1 \approx a_2$. Let $b_1, b_2 \in \mathfrak{M}_\sigma$ with $b_1 \sim b_2$. By (ii) we have $a_1 \sim a_1$, so $a_1 b_1 \sim a_1 b_2$. Also $a_1 b_2 \approx a_2 b_2$ because $a_1 \approx a_2$. And so $a_1 b_2 \sim a_2 b_2$ by induction. Therefore $a_1 b_1 \sim a_2 b_2$. So $a_1 \sim a_2$.

- Now suppose $a_1 \sim a_2$. Let $b \in \mathfrak{M}_\sigma$. Then $b \sim b$ by (ii). So $a_1 b \sim a_2 b$ and so $a_1 b \approx a_2 b$ by induction. Therefore $a_1 \approx a_2$. □

By the previous two lemmas we obtain the following proposition.

2.4.10. PROPOSITION. *The following are equivalent.*

(i) *$\mathfrak{M}$ is quasi-extensional.*

(ii) $\forall \sigma \in \mathbb{T} \forall a_1, a_2 \in \mathfrak{M}_\sigma [\, a_1 \approx a_2 \Longleftrightarrow a_1 \sim a_2 \,]$.

(iii) $\forall \sigma \in \mathbb{T} \forall a \in \mathfrak{M}_\sigma [\, a \sim a \,]$. □

2.4.11. THEOREM. *$\mathfrak{M}$ is quasi-extensional if and only if $\mathfrak{M}^Z = \mathfrak{M}$.*

PROOF. (i) Let $\mathfrak{M}$ be quasi-extensional. Then for all $\sigma \in \mathbb{T}$ and all $a \in \mathfrak{M}_\sigma$ we have $a \sim a$ and so $\mathfrak{M}^Z_\sigma = \mathfrak{M}_\sigma$. Therefore $\mathfrak{M}^Z = \mathfrak{M}$.

(ii) If $\mathfrak{M}^Z = \mathfrak{M}$, then $\mathfrak{M}$ is quasi-extensional, because $\mathfrak{M}^Z$ is quasi-extensional. $\Box$

Next it will be shown that if $\mathfrak{M}$ is quasi-combinatory, then $\mathfrak{M}$ has the Zucker Property and $\mathfrak{M}^Z$ is also quasi-combinatory, and therefore $\mathfrak{M}^Z$ is a λ-structure.

2.4.12. LEMMA. *If $\mathfrak{M}$ is quasi-combinatory, then $\mathfrak{M}$ has the Zucker Property.*

PROOF. Let $\mathfrak{M}$ be quasi-combinatory and let $(\mathbf{k}_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}$ be a quasi-k-family in $\mathfrak{M}$. We show by induction on the structure of σ that

$$\forall\sigma\in\mathbb{T}\exists a\in\mathfrak{M}_\sigma[\, a\sim a\,].$$

(i) For $\sigma=0$ take any $a\in\mathfrak{M}_0$; then $a\sim a$.

(ii) Let $\sigma,\tau\in\mathbb{T}$. By induction there exists $a\in\mathfrak{M}_\tau$ with $a\sim a$. Then $\mathbf{k}_{\tau,\sigma}a\in\mathfrak{M}_{\sigma\to\tau}$. We will show that $\mathbf{k}_{\tau,\sigma}a\sim\mathbf{k}_{\tau,\sigma}a$. Let $b_1,b_2\in\mathfrak{M}_\sigma$ with $b_1\sim b_2$. Then $\mathbf{k}_{\tau,\sigma}ab_1\approx a\approx\mathbf{k}_{\tau,\sigma}ab_2$, so $\mathbf{k}_{\tau,\sigma}ab_1\approx\mathbf{k}_{\tau,\sigma}ab_2$ and so $\mathbf{k}_{\tau,\sigma}ab_1\sim\mathbf{k}_{\tau,\sigma}ab_2$. Therefore $\mathbf{k}_{\tau,\sigma}a\sim\mathbf{k}_{\tau,\sigma}a$.

This shows that $\mathfrak{M}$ has the Zucker Property. $\Box$

2.4.13. LEMMA. *Let $\mathfrak{M}$ be quasi-combinatory.*

(i) *Every quasi-k-family in $\mathfrak{M}$ is a quasi-k-family in $\mathfrak{M}^Z$.*

(ii) *Every quasi-s-family in $\mathfrak{M}$ is a quasi-s-family in $\mathfrak{M}^Z$.*

PROOF. Straightforward, using that $a\approx b\Rightarrow a\sim b$ for all objects a and b of $\mathfrak{M}^Z$. $\Box$

2.5. Term algebras and term models.

Every system Σ gives rise to a type structure $\mathfrak{M}(\Sigma)$ called the term structure of Σ. The objects of $\mathfrak{M}(\Sigma)$ are equivalence classes of Σ-terms under provable equality. If Σ is a β-system, then there exists a canonical Σ-interpretation $[\![\;]\!]$ in $\mathfrak{M}(\Sigma)$ such that $(\mathfrak{M}(\Sigma),[\![\;]\!])$ is a Σ-model. This Σ-model is called the (open) term model of Σ.

2.5.1. DEFINITION. (i) For all $\sigma\in\mathbb{T}$ and $M\in\mathcal{T}_\sigma(\Sigma)$

$$[M]=\{N\in\mathcal{T}_\sigma(\Sigma)\,|\,\Sigma\vdash M=N\,\}.$$

(ii) The (open) term structure of Σ is

$$\mathfrak{M}(\Sigma)=((\mathfrak{M}_\sigma(\Sigma))_{\sigma\in\mathbb{T}},(App_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}})$$

where $App_{\sigma,\tau}:\mathfrak{M}_{\sigma\to\tau}(\Sigma)\times\mathfrak{M}_\sigma(\Sigma)\to\mathfrak{M}_\tau(\Sigma)$ is defined by $App_{\sigma,\tau}([M],[N])=[MN]$. $\Box$

Let Σ be a β-system. Define $[\![\,]\!] : \mathcal{T}(\Sigma) \times Val(\mathfrak{M}(\Sigma)) \to |\mathfrak{M}(\Sigma)|$ as follows. Let $M \in \mathcal{T}(\Sigma)$ and $\rho \in Val(\mathfrak{M}(\Sigma))$. Let $\vec{x} = x_1, \ldots, x_n$ with $FV(M) \subseteq \{\vec{x}\}$ and let $\rho(x_i) = [P_i]$ $(i = 1, \ldots, n)$. Then define $[\![M]\!]_\rho = [M[\vec{x} := \vec{P}]]$. Then $[\![\,]\!]$ is well-defined, because

$$\Sigma \vdash P_i = Q_i \ (i = 1, \ldots, n) \Rightarrow \Sigma \vdash M[\vec{x} := \vec{P}] = M[\vec{x} := \vec{Q}].$$

This can be shown by induction on the structure of M, or by using that Σ is a β-system. Using that Σ is a β-system, it can easily be shown that $[\![\,]\!]$ is a Σ-interpretation in $\mathfrak{M}$.

2.5.2. THEOREM. $(\mathfrak{M}(\Sigma), [\![\,]\!])$ *is a* Σ-*model.*

PROOF. We will show $Th(\mathfrak{M}(\Sigma), [\![\,]\!]) = Th(\Sigma)$.

(i) Let $(M = N) \in Th(\mathfrak{M}(\Sigma), [\![\,]\!])$. Define $\rho \in Val(\mathfrak{M}(\Sigma))$ by $\rho(x) = [x]$. Then $[M] = [\![M]\!]_\rho = [\![N]\!]_\rho = [N]$, so $\Sigma \vdash M = N$ and so $(M = N) \in Th(\Sigma)$.

(ii) Let $(M = N) \in Th(\Sigma)$. Then $\Sigma \vdash M = N$. Let $\rho \in Val(\mathfrak{M}(\Sigma))$. Let $\vec{x} = x_1, \ldots, x_n$ with $FV(M) \cup FV(N) \subseteq \{\vec{x}\}$ and $\rho(x_i) = [P_i]$ $(i - 1, \ldots, n)$. Then, in Σ we have

$$M[\vec{x} := \vec{P}] = (\lambda\vec{x}.M)\vec{P} = (\lambda\vec{x}.N)\vec{P} = N[\vec{x} := \vec{P}].$$

So $[\![M]\!]_\rho = [M[\vec{x} := \vec{P}]] = [N[\vec{x} := \vec{P}]] = [\![N]\!]_\rho$, so $\mathfrak{M}(\Sigma), [\![\,]\!] \models M = N$ and so $(M = N) \in Th(\mathfrak{M}(\Sigma), [\![\,]\!])$.

From (i) and (ii) it follows that $Th(\mathfrak{M}(\Sigma), [\![\,]\!]) = Th(\Sigma)$, therefore $Th(\mathfrak{M}(\Sigma), [\![\,]\!])$ is a Σ-theory, so $(\mathfrak{M}(\Sigma), [\![\,]\!])$ is a Σ-model. $\square$

If Σ is such that $\mathcal{T}_\sigma(\Sigma)^0 \neq \emptyset$ for every $\sigma \in \mathbb{T}$ (or equivalently: $\mathcal{T}_0(\Sigma)^0 \neq \emptyset$), then there exists a type structure $\mathfrak{M}^0(\Sigma)$ of closed Σ-terms modulo provable equality. And again, if Σ is a β-system, there exists a Σ-interpretation $[\![\,]\!]$ in $\mathfrak{M}^0(\Sigma)$ such that $(\mathfrak{M}^0(\Sigma), [\![\,]\!])$ is a Σ-algebra. In general $(\mathfrak{M}^0(\Sigma), [\![\,]\!])$ is not a Σ-model. The Σ-algebra $(\mathfrak{M}^0(\Sigma), [\![\,]\!])$ is called the closed term algebra of Σ.

Let Σ be such that $\mathcal{T}_0(\Sigma)^0 \neq \emptyset$. Then $\mathcal{T}_\sigma(\Sigma)^0 \neq \emptyset$ for every $\sigma \in \mathbb{T}$.

2.5.3. DEFINITION. (i) For every $\sigma \in \mathbb{T}$ and $M \in \mathcal{T}_\sigma(\Sigma)^0$

$$[M]^0 = \{N \in \mathcal{T}_\sigma(\Sigma)^0 \mid \Sigma \vdash M = N\}.$$

(ii) $\mathfrak{M}^0_\sigma = \{[M]^0 \mid M \in \mathcal{T}_\sigma(\Sigma)^0\}$.

(iii) The closed term structure of Σ is

$$\mathfrak{M}^0(\Sigma) = ((\mathfrak{M}^0_\sigma(\Sigma))_{\sigma\in\mathbb{T}}, (App_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}})$$

where $App_{\sigma,\tau} : \mathfrak{M}^0_{\sigma\to\tau}(\Sigma) \times \mathfrak{M}^0_\sigma(\Sigma) \to \mathfrak{M}^0_\tau(\Sigma)$ is defined by $App_{\sigma,\tau}([M]^0, [N]^0) = [MN]^0$. $\square$

Suppose Σ is a β-system. Define $[\![\,]\!] : \mathcal{T}(\Sigma) \times Val(\mathfrak{M}^0(\Sigma)) \to |\mathfrak{M}^0(\Sigma)|$ as follows. Let $M \in \mathcal{T}(\Sigma)$ and $\rho \in Val(\mathfrak{M}^0(\Sigma))$. Let $\vec{x} = x_1, \ldots, x_n$ be such that $FV(M) \subseteq \{\vec{x}\}$. Let $\rho(x_i) = [P_i]$ $(i = 1, \ldots, n)$. Now define $[\![M]\!]_\rho = [M[\vec{x} := \vec{P}]]^0$ (note that $M[\vec{x} := \vec{P}]$ is closed). Using that Σ is a β-system it can easily be shown that $[\![\,]\!]$ is a Σ-interpretation in $\mathfrak{M}^0(\Sigma)$.

2.5.4. THEOREM. *$(\mathfrak{M}^0(\Sigma), [\![\,]\!])$ is a Σ-algebra.*

PROOF. Let $\Sigma \vdash M = N$. Let $\rho \in Val(\mathfrak{M}^0(\Sigma))$. Let $\vec{x} = x_1, \ldots, x_n$ be such that $FV(M) \cup FV(N) \subseteq \{\vec{x}\}$ and say $\rho(x_i) = [P_i]$ for $i = 1, \ldots, n$. Then in Σ we have

$$M[\vec{x} := \vec{P}] = (\lambda\vec{x}.M)\vec{P} = (\lambda\vec{x}.N)\vec{P} = N[\vec{x} := \vec{P}]$$

so $[\![M]\!]_\rho = [M[\vec{x} := \vec{P}]] = [N[\vec{x} := \vec{P}]] = [\![N]\!]_\rho$ and therefore $\mathfrak{M}^0(\Sigma), [\![\,]\!] \models M = N$. So $(\mathfrak{M}^0(\Sigma), [\![\,]\!])$ is a Σ-algebra. $\square$

In the next section it will be shown that $(\mathfrak{M}^0(\Sigma), [\![\,]\!])$ is in general not a Σ-model.

2.6. Comparing models and quasi-models.

In section 2.2 we saw that for every system Σ we have

- Σ-model $\Rightarrow$ Σ-algebra
- quasi-Σ-model $\Rightarrow$ quasi-Σ-algebra
- Σ-algebra $\Rightarrow$ quasi-Σ-algebra
- Σ-model $\Rightarrow$ quasi-Σ-algebra.

In this section we will show that these implications are the only ones that hold in general between the notions of model, algebra, quasi-model and quasi-algebra. For every non-implication we will construct a system Σ and a quasi-Σ-structure that proves this non-implication.

The first non-implications follow by looking at the term model $(\mathfrak{M}(\lambda^{\to}), [\![\,]\!])$ of $\lambda^{\to}$. Note that $(\mathfrak{M}(\lambda^{\to}), [\![\,]\!])$ is a $\lambda^{\to}$-model, a $\lambda^{\to}$-algebra and a quasi-$\lambda^{\to}$-algebra.

2.6.1. THEOREM. *$(\mathfrak{M}(\lambda^{\to}), [\![\,]\!])$ is not a quasi-$\lambda^{\to}$-model.*

PROOF. Suppose that $(\mathfrak{M}(\lambda^{\to}), [\![\,]\!])$ is a quasi-$\lambda^{\to}$-model. Let $x \in Var_{0\to 0}$, $y \in Var_0$ and $z \in Var_{((0\to 0)\to(0\to 0))\to 0}$. First we show

$$\mathfrak{M}(\lambda^{\to}), [\![\,]\!] \models_q \lambda xy.xy = \lambda x.x.$$

Let $\rho \in Val(\mathfrak{M}(\lambda^{\rightarrow}))$. Then $[\![\lambda xy.xy]\!]_\rho = [\lambda xy.xy]$ and $[\![\lambda x.x]\!] = [\lambda x.x]$. Let $a \in \mathfrak{M}_{0\rightarrow 0}(\lambda^{\rightarrow})$ and $b \in \mathfrak{M}_0(\lambda^{\rightarrow})$. Say $a = [P]$ and $b = [Q]$. Then

$$\begin{aligned} [\lambda xy.xy]ab &= [(\lambda xy.xy)PQ] \\ &= [PQ] \\ &= [(\lambda x.x)PQ] \\ &= [\lambda x.x]ab. \end{aligned}$$

So $[\lambda xy.xy] \approx [\lambda x.x]$. Therefore

$$\mathfrak{M}(\lambda^{\rightarrow}), [\![\ ]\!] \models_q \lambda xy.xy = \lambda x.x.$$

By the assumption that $(\mathfrak{M}(\lambda^{\rightarrow}), [\![\ ]\!])$ is a quasi-$\lambda^{\rightarrow}$-model, it follows that the set $Th_q(\mathfrak{M}(\lambda^{\rightarrow}), [\![\ ]\!])$ is a $\lambda^{\rightarrow}$-theory and therefore

$$\mathfrak{M}(\lambda^{\rightarrow}), [\![\ ]\!] \models_q z(\lambda xy.xy) = z(\lambda x.x).$$

Let $\rho \in Val(\mathfrak{M}(\lambda^{\rightarrow}))$ with $\rho(z) = [z]$. Then $[z(\lambda xy.xy)] = [\![z(\lambda xy.xy)]\!]_\rho \approx [\![z(\lambda x.x)]\!]_\rho = [z(\lambda x.x)]$, but $[z(\lambda xy.xy)]$ and $[z(\lambda x.x)]$ are type-0 objects of $\mathfrak{M}(\lambda^{\rightarrow})$, so $[z(\lambda xy.xy)] = [z(\lambda x.x)]$, so $z(\lambda xy.xy) =_\beta z(\lambda x.x)$. But $z(\lambda xy.xy)$ and $z(\lambda x.x)$ are β-normal forms, so by $\mathbf{CR}(\beta)$ we have $z(\lambda xy.xy) \equiv z(\lambda x.x)$. Contradiction. $\Box$

So $(\mathfrak{M}(\lambda^{\rightarrow}), [\![\ ]\!])$ is a $\lambda^{\rightarrow}$-model, a $\lambda^{\rightarrow}$-algebra and a quasi-$\lambda^{\rightarrow}$-algebra but not a quasi-$\lambda^{\rightarrow}$-model, and therefore we have the following non-implications:

- model $\not\Rightarrow$ quasi-model
- algebra $\not\Rightarrow$ quasi-model
- quasi-algebra $\not\Rightarrow$ quasi-model.

Next the system $\lambda\mathsf{ABC}$ will be defined. This system contains 'Plotkin-terms' of a very simple form, and from this it follows that the closed term algebra of $\lambda\mathsf{ABC}$ is not a $\lambda\mathsf{ABC}$-model.

2.6.2. Definition. The system $\lambda\mathsf{ABC}$ is the $\beta\delta$-system defined as follows.

(i) The constants of $\lambda\mathsf{ABC}$, their types and their arities are given by the following table.

constant	type	arity
A	$0\rightarrow 0$	1
B	$0\rightarrow 0$	1
C	0	0

(ii) Let δ be the reductor on $\mathcal{T}(\lambda\mathsf{ABC})$ consisting of the following reduction rules:

(1) $\mathbf{AC} \to \mathbf{C}$

(2) $\mathbf{BC} \to \mathbf{C}$.

(iii) $\lambda\mathsf{ABC} = (\mathcal{T}(\lambda\mathsf{ABC}), \beta\delta)$. □

2.6.3. Lemma. $\lambda\mathsf{ABC} \models \mathbf{CR}$.

Proof. By the First Church-Rosser Theorem. □

2.6.4. Proposition. $\forall M \in \mathcal{T}_0(\lambda\mathsf{ABC})^0[\, M \twoheadrightarrow_{\beta\delta} \mathbf{C}\,]$.

Proof. Let $\mathbf{NF}_0(\beta)^0$ denote the set of all closed β-normal forms of type 0 in $\mathcal{T}(\lambda\mathsf{ABC})$. By considering the syntactic structure of the terms in $\mathbf{NF}_0(\beta)^0$ it can be shown that

$$\mathbf{NF}_0(\beta)^0 = \{\mathbf{C}, \mathbf{AC}, \mathbf{BC}, \mathbf{A}(\mathbf{AC}), \mathbf{B}(\mathbf{AC}), \mathbf{A}(\mathbf{BC}), \ldots\}$$

and therefore we have

$$\forall N \in \mathbf{NF}_0(\beta)^0[\, N \twoheadrightarrow_{\delta} \mathbf{C}\,].$$

Now let $M \in \mathcal{T}_0(\lambda\mathsf{ABC})^0$. Because $\mathbf{SN}(\beta)$ there exists $N \in \mathbf{NF}_0(\beta)^0$ such that $M \twoheadrightarrow_{\beta} N$. Then $N \twoheadrightarrow_{\delta} \mathbf{C}$ and so $M \twoheadrightarrow_{\beta\delta} \mathbf{C}$. □

Let $(\mathfrak{M}^0(\lambda\mathsf{ABC}), [\![\,]\!])$ be the closed term algebra of $\lambda\mathsf{ABC}$. Then $(\mathfrak{M}^0(\lambda\mathsf{ABC}), [\![\,]\!])$ is also a quasi-$\lambda\mathsf{ABC}$-algebra. We are going to show that $(\mathfrak{M}^0(\lambda\mathsf{ABC}), [\![\,]\!])$ is not a $\lambda\mathsf{ABC}$-model.

2.6.5. Proposition. (i) $\mathfrak{M}^0(\lambda\mathsf{ABC}), [\![\,]\!] \models \mathbf{A}x = \mathbf{B}x$.

(ii) $\mathfrak{M}^0(\lambda\mathsf{ABC}), [\![\,]\!] \not\models \lambda x.\mathbf{A}x = \lambda x.\mathbf{B}x$.

Proof. (i) Let $P \in \mathcal{T}_0(\lambda\mathsf{ABC})^0$. Then $P \twoheadrightarrow_{\beta\delta} \mathbf{C}$, by 2.6.4. So

$$\mathbf{A}P =_{\beta\delta} \mathbf{AC} =_{\beta\delta} \mathbf{C} =_{\beta\delta} \mathbf{BC} =_{\beta\delta} \mathbf{B}P.$$

Therefore $\lambda\mathsf{ABC} \vdash (\mathbf{A}x)[x := P] = (\mathbf{B}x)[x := P]$, and so

$$\mathfrak{M}^0(\lambda\mathsf{ABC}), [\![\,]\!] \models \mathbf{A}x = \mathbf{B}x.$$

(ii) Suppose $\mathfrak{M}^0(\lambda\mathsf{ABC}), [\![\,]\!] \models \lambda x.\mathbf{A}x = \lambda x.\mathbf{B}x$. Then $\lambda\mathsf{ABC} \vdash \lambda x.\mathbf{A}x = \lambda x.\mathbf{B}x$, because $\lambda x.\mathbf{A}x$ and $\lambda x.\mathbf{B}x$ are closed. Then $\lambda x.\mathbf{A}x =_{\beta\delta} \lambda x.\mathbf{B}x$. But $\lambda x.\mathbf{A}x$ and $\lambda x.\mathbf{B}x$ are different β-normal forms. This contradicts $\mathbf{CR}(\beta\delta)$. □

2.6.6. Corollary. $(\mathfrak{M}^0(\lambda\mathsf{ABC}), [\![\,]\!])$ *is not a* $\lambda\mathsf{ABC}$-*model.* □

This gives us the following non-implications:

- algebra $\not\Rightarrow$ model.

- quasi-algebra $\not\Rightarrow$ model.

The type structure **HRO** comes from partial recursive application and was introduced in [Troelstra 1973]. In the following we will give a definition of **HRO** and show that it has the Zucker Property. After that it will be shown that the Zucker collapse $\mathbf{HRO}^Z$ of **HRO** can be made into a quasi-$\lambda^{\rightarrow}$-model which is not a $\lambda^{\rightarrow}$-algebra, thereby proving the remaining three non-implications.

2.6.7. DEFINITION. (i) If $m \in \omega$, then $\mathbb{P}_m(\omega)$ denotes the set of all partial m-ary functions on ω.

(ii) $pREC$ denotes the set of all partial recursive functions on ω.

(iii) If $m \in \omega$, then $pREC_m$ denotes the set of all m-ary partial recursive functions on ω.

(iv) Let $e, m \in \omega$. Then $\{e\}^m$ is the e-th m-ary partial recursive function on ω. $\square$

The following two theorems are borrowed from recursion theory.

2.6.8. THEOREM (Enumeration Theorem).
For every $m \in \omega$ there exists an enumeration function $\mathcal{E}_m \in pREC_{m+1}$ such that for all $e \in \omega$ and $\vec{a} \in \omega^m$

$$\mathcal{E}_m(e, \vec{a}) \simeq \{e\}^m(\vec{a}). \ \square$$

2.6.9. THEOREM (Parameter Theorem; s-m-n-Theorem).
Let $m, n \in \omega$ with $m, n > 0$. Then there exists an injective, recursive function $S^m_n : \omega^{m+1} \to \omega$ such that for all $e \in \omega$, $\vec{a} \in \omega^m$ and $\vec{b} \in \omega^n$

$$\{S^m_n(e, \vec{a})\}^n(\vec{b}) \simeq \{e\}^{m+n}(\vec{a}, \vec{b}). \ \square$$

2.6.10. DEFINITION. (i) Partial recursive application is defined by

$$a \cdot b \simeq \{a\}^1(b)$$

for all $a, b \in \omega$.

(ii) $\mathcal{P} = (\omega, \cdot)$. $\square$

$\mathcal{P}$ is a partial applicative structure that is sometimes called Kleene's partial applicative structure. For partial application we also use association to the left, and for $a, b \in \omega$ we will often write ab instead of $a \cdot b$.

In $\mathcal{P}$ we can represent partial functions on ω.

2.6.11. DEFINITION. Let $m \in \omega$ and $\varphi \in \mathbb{P}_{m+1}(\omega)$.

(i) Let $f \in \omega$. Then f $\mathcal{P}$-represents φ if for all $a_1, \dots, a_{m+1} \in \omega$

(1) $f a_1 \dots a_m \downarrow$

(2) $\varphi(a_1,\ldots,a_{m+1}) \simeq fa_1\ldots a_{m+1}$.

(ii) φ is $\mathcal{P}$-representable if there exists $f \in \omega$ such that f $\mathcal{P}$-represents φ. □

2.6.12. DEFINITION. Let $m \in \omega$ and $f \in \omega$.

(i) f is m-total if for all $a_1,\ldots,a_m \in \omega$ $fa_1\ldots a_m \downarrow$.

(ii) f is m-injective if f is m-total and for all $\vec{a},\vec{b} \in \omega^m$

$$fa_1\ldots a_m = fb_1\ldots b_m \Rightarrow a_i = b_i \; (i = 1,\ldots,m). \; \Box$$

2.6.13. DEFINITION. Let $m \in \omega$ and $\varphi \in \mathbb{P}_{m+1}(\omega)$.

(i) Let $f \in \omega$. Then f m-injectively $\mathcal{P}$-represents φ if f is m-injective and f $\mathcal{P}$-represents φ.

(ii) φ is m-injectively $\mathcal{P}$-representable if there exists an m-injective $f \in \omega$ that $\mathcal{P}$-represents φ. □

2.6.14. PROPOSITION. *For every $m \in \omega$ and every $\varphi \in pREC_{m+1}$ we have that φ is m-injectively $\mathcal{P}$-representable.*

PROOF. By induction on m.

(i) Let $\varphi \in pREC_1$. Then $\varphi = \{e\}^1$ for some $e \in \omega$. Then for all $a \in \omega$

$$\varphi(a) \simeq \{e\}^1(a) \simeq ea.$$

Note that e is 0-total and 0-injective. So φ is 0-injectively $\mathcal{P}$-representable.

(ii) Let $\varphi \in pREC_{m+2}$. Then $\varphi = \{e\}^{m+2}$ for some $e \in \omega$. Then for all $a_1,\ldots,a_{m+2} \in \omega$

$$\begin{aligned}\varphi(a_1,\ldots,a_{m+2}) &\simeq \{e\}^{m+2}(a_1,\ldots,a_{m+2})\\ &\simeq \{S_1^{m+1}(e,a_1,\ldots,a_{m+1})\}^1(a_{m+2})\\ &\simeq S_1^{m+1}(e,a_1,\ldots,a_{m+1})\cdot a_{m+2}.\end{aligned}$$

Define $\psi \in \mathbb{P}_{m+1}(\omega)$ by

$$\psi(a_1,\ldots,a_{m+1}) = S_1^{m+1}(e,a_1,\ldots,a_{m+1}).$$

Then ψ is recursive, so by induction ψ is m-injectively $\mathcal{P}$-representable. Let $f \in \omega$ $\mathcal{P}$-represent ψ. Then f also $\mathcal{P}$-represents φ, and by the injectivity of S_1^{m+1} we have that f is $(m+1)$-injective. Therefore φ is $(m+1)$-injectively $\mathcal{P}$-representable. □

2.6.15. LEMMA. *For all $m \in \omega$ and $\varphi \in \mathbb{P}_{m+1}(\omega)$: if φ is $\mathcal{P}$-representable, then φ is partial recursive.*

PROOF. By induction on m.

(i) Let $\varphi \in \mathbb{P}_1(\omega)$ be $\mathcal{P}$-representable. Let $f \in \omega$ $\mathcal{P}$-represent φ. Then $\varphi = \{f\}^1$ and so φ is partial recursive.

(ii) Let $\varphi \in \mathbb{P}_{m+2}(\omega)$ be $\mathcal{P}$-representable. Let $f \in \omega$ $\mathcal{P}$-represent φ. Then for all $a_1, \ldots, a_{m+2} \in \omega$

$$\begin{aligned} \varphi(a_1, \ldots, a_{m+2}) &\simeq fa_1 \ldots a_{m+2} \\ &\simeq (fa_1 \ldots a_{m+1})a_{m+2} \\ &\simeq \{fa_1 \ldots a_{m+1}\}^1(a_{m+2}) \\ &\simeq \mathcal{E}_1(fa_1 \ldots a_{m+1}, a_{m+2}). \end{aligned}$$

Define $\psi \in \mathbb{P}_{m+1}(\omega)$ by

$$\psi(a_1, \ldots, a_{m+1}) \simeq fa_1 \ldots a_{m+1}.$$

Then f $\mathcal{P}$-represents ψ, because f is $(m+1)$-total and therefore also m-total. So by induction $\psi \in pREC$ and so $\varphi \in pREC$ because

$$\varphi(a_1, \ldots, a_{m+2}) \simeq \mathcal{E}_1(\psi(a_1, \ldots, a_{m+1}), a_{m+2}). \ \Box$$

2.6.16. PROPOSITION. (i) *There exists* $\mathbf{k} \in \omega$ *such that for all* $a, b \in \omega$

$$\mathbf{k}ab \simeq a.$$

(ii) *There exists a 2-injective* $\mathbf{s} \in \omega$ *such that for all* $a, b, c \in \omega$

$$\mathbf{s}abc \simeq ac(bc).$$

PROOF. (i) Define $\varphi \in \mathbb{P}_2(\omega)$ by $\varphi(a, b) = a$. Then $\varphi \in pREC$, so φ is $\mathcal{P}$-representable. Let $\mathbf{k} \in \omega$ be such that $\mathbf{k}$ $\mathcal{P}$-represents φ.

(ii) Define $\psi \in \mathbb{P}_3(\omega)$ by $\psi(a, b, c) \simeq ac(bc)$. Then for all $a, b, c \in \omega$

$$\psi(a, b, c) \simeq \mathcal{E}_1(\mathcal{E}_1(a, c), \mathcal{E}_1(b, c)).$$

So ψ is partial recursive and so ψ is 2-injectively $\mathcal{P}$-representable. Let $\mathbf{s} \in \omega$ be such that $\mathbf{s}$ 2-injectively $\mathcal{P}$-represents ψ. $\Box$

In the following let $\mathbf{k}, \mathbf{s} \in \omega$ with the properties described in 2.6.16. By the 2-injectivity of $\mathbf{s}$ we have $\mathbf{s}ab\!\downarrow$ and

$$\mathbf{s}ab = \mathbf{s}a'b' \Rightarrow a = a' \ \& \ b = b'.$$

The structure $(\omega, \cdot, \mathbf{k}, \mathbf{s})$ is called a partial combinatory algebra. For more on partial combinatory algebras, see [Bethke 1988].

2.6.17. DEFINITION. $\mathbf{i} = \mathbf{skk}$. $\Box$

2.6.18. LEMMA. (i) *For all* $a \in \omega$ $\mathbf{i}a = a$.
(ii) $\mathbf{ki} \neq \mathbf{k}$.
(iii) $\mathbf{s}(\mathbf{ki})\mathbf{i} \neq \mathbf{i}$.

PROOF. (i) $\mathbf{i}a = \mathbf{skk}a = \mathbf{k}a(\mathbf{k}a) = a$.
(ii) Suppose $\mathbf{ki} = \mathbf{k}$. Let $a, b \in \omega$. Then $\mathbf{ki}ab = \mathbf{k}ab$. But $\mathbf{ki}ab = b$ and $\mathbf{k}ab = a$, so $a = b$. Contradiction.
(iii) By (ii) and the 2-injectivity of $\mathbf{s}$. $\square$

Next the type structure **HRO** will be defined.

2.6.19. DEFINITION. (i) For every $\sigma \in \mathbb{T}$ define $\mathbf{HRO}_\sigma \subseteq \omega$ by induction on the structure of σ as follows.

(1) $\mathbf{HRO}_0 = \omega$.

(2) $\mathbf{HRO}_{\sigma\to\tau} = \{a \in \omega | \forall b \in \mathbf{HRO}_\sigma [ab\downarrow \ \& \ ab \in \mathbf{HRO}_\tau]\}$.

(ii) Let $App_{\sigma,\tau} : \mathbf{HRO}_{\sigma\to\tau} \times \mathbf{HRO}_\sigma \to \mathbf{HRO}_\tau$ be defined by

$$App_{\sigma,\tau}(a, b) = ab.$$

(iii) $\mathbf{HRO} = ((\mathbf{HRO}_\sigma)_{\sigma\in\mathbb{T}}, (App_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}})$. $\square$

It is easy to show that

$$\mathbf{k} \in \mathbf{HRO}_{\sigma\to\tau\to\sigma}$$

$$\mathbf{s} \in \mathbf{HRO}_{(\sigma\to\tau\to\upsilon)\to(\sigma\to\tau)\to\sigma\to\upsilon}$$

for all $\sigma, \tau, \upsilon \in \mathbb{T}$. Using $\mathbf{k}$ it is then easy to see that $\mathbf{HRO}_\sigma \neq \emptyset$ for every $\sigma \in \mathbb{T}$, so that **HRO** is indeed a type structure.

For all $\sigma, \tau, \upsilon \in \mathbb{T}$ define $\mathbf{k}_{\sigma,\tau} = \mathbf{k}$ and $\mathbf{s}_{\sigma,\tau,\upsilon} = \mathbf{s}$. Then $(\mathbf{k}_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}$ is a quasi-k-family (even a k-family) in **HRO**, and $(\mathbf{s}_{\sigma,\tau,\upsilon})_{\sigma,\tau,\upsilon\in\mathbb{T}}$ is a quasi-s-family (even an s-family) in **HRO**, so **HRO** is quasi-combinatory (and even combinatory) and therefore **HRO** has the Zucker Property. Applying the Zucker collapse to **HRO** yields the quasi-extensional type structure $\mathbf{HRO}^Z$. Then $(\mathbf{k}_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}$ is a quasi-k-family in $\mathbf{HRO}^Z$ and $(\mathbf{s}_{\sigma,\tau,\upsilon})_{\sigma,\tau,\upsilon\in\mathbb{T}}$ is a quasi-s-family in $\mathbf{HRO}^Z$. So $\mathbf{HRO}^Z$ is a λ-structure.

Let $\mathbb{C} = (\mathbb{C}_\sigma)_{\sigma\in\mathbb{T}}$ be such that $\mathbb{C}_\sigma = \emptyset$ for all $\sigma \in \mathbb{T}$. Let $(\ )$ be the $\mathbf{CL}(\mathbb{C}^*)$-interpretation in $\mathbf{HRO}^Z$ such that

$$(\mathsf{K}_{\sigma,\tau})_\rho = \mathbf{k}_{\sigma,\tau}$$

$$(\mathsf{S}_{\sigma,\tau,\upsilon})_\rho = \mathbf{s}_{\sigma,\tau,\upsilon}.$$

Note that $\mathcal{T}(\mathbb{C}) = \mathcal{T}$. Let $[\![\ ]\!]$ be the quasi-$\mathcal{T}$-interpretation in $\mathbf{HRO}^Z$ with $[\![M]\!]_\rho = (M_{CL})_\rho$. Then $[\![\ ]\!]$ is even a $\mathcal{T}$-interpretation in $\mathbf{HRO}^Z$. By the quasi-extensionality of $\mathbf{HRO}^Z$ we obtain the following theorem.

2.6.20. THEOREM. $(\mathbf{HRO}^Z, [\![\,]\!])$ *is a quasi-*$\lambda^{\rightarrow}$*-model and a quasi-*$\lambda^{\rightarrow}$*-algebra.* $\Box$

Next it will be shown that $(\mathbf{HRO}^Z, [\![]\!])$ is not a $\lambda^{\rightarrow}$-algebra and therefore not a $\lambda^{\rightarrow}$-model. Let $x, y \in Var_0$. Then $\lambda^{\rightarrow} \vdash \lambda x.(\lambda y.y)x = \lambda x.x$. But $[\![\lambda x.(\lambda y.y)x]\!] = \mathbf{s}(\mathbf{ki})\mathbf{i}$ and $[\![\lambda x.x]\!] = \mathbf{i}$ and $\mathbf{s}(\mathbf{ki})\mathbf{i} \neq \mathbf{i}$, so $[\![\lambda x.(\lambda y.y)x]\!] \neq [\![\lambda x.x]\!]$.

2.6.21. THEOREM. $(\mathbf{HRO}^Z, [\![\,]\!])$ *is not a* $\lambda^{\rightarrow}$*-algebra and not a* $\lambda^{\rightarrow}$*-model.* $\Box$

These two theorems provide us with the last three non-implications.

- quasi-model $\not\Rightarrow$ model
- quasi-algebra $\not\Rightarrow$ algebra
- quasi-model $\not\Rightarrow$ algebra.

Chapter 3

Models in categories

Cartesian closed categories provide a systematic method for constructing algebras and models for simply typed λ-calculi. For every cartesian closed category $\mathcal{C}$ and every non-empty object a of $\mathcal{C}$ we will construct a type structure $\mathfrak{M}_{\mathcal{C}}(a)$ and a $\lambda^{\rightarrow}$-interpretation $[\![\ ]\!]$ in $\mathfrak{M}_{\mathcal{C}}(a)$ such that $(\mathfrak{M}_{\mathcal{C}}(a), [\![\ ]\!])$ is a $\lambda^{\rightarrow}\eta$-algebra. If $\mathcal{C}$ is λ-extensional, then $(\mathfrak{M}_{\mathcal{C}}(a), [\![\ ]\!])$ is even a $\lambda^{\rightarrow}\eta$-model. So we have a construction method for $\lambda^{\rightarrow}\eta$-algebras with two parameters: a cartesian closed category $\mathcal{C}$ and a non-empty object in $\mathcal{C}$. By choosing these parameters carefully one can obtain algebras and models for stronger systems than $\lambda^{\rightarrow}\eta$.

3.1. Cartesian closed categories.

3.1.1. Definition. A category $\mathcal{C}$ consists of

(i) a class $O = O(\mathcal{C})$ of objects; for objects we will use $a, b, c, \ldots$

(ii) a class $M = M(\mathcal{C})$ of morphisms; for morphisms we will use $f, g, h, \ldots$

(iii) two mappings $dom : M \to O$ and $cod : M \to O$; if $f \in M$, then the object $dom(f)$ is called the domain of f and the object $cod(f)$ is called the codomain of f. If $f \in M$, $a = dom(f)$ and $b = cod(f)$, then we write $f : a \to b$.

(iv) for any two morphisms $f : a \to b$ and $g : b \to c$ we have a morphism $g \circ f : a \to c$ called the composition of f and g.

(v) for every object a of $\mathcal{C}$ we have a morphism $1_a : a \to a$ called the identity on a.

A structure $\mathcal{C}$ consisting of the items (i)-(v) is a category if it has the Associativity Property and the Identity Property described below.

1. Associativity Property: for all objects a, b, c, d and all morphisms $f : a \to b$, $g : b \to c$ and $h : c \to d$ we have
$$(h \circ g) \circ f = h \circ (g \circ f).$$
2. Identity Property: for all objects a and b and every morphism $f : a \to b$ we have
$$f \circ 1_a = f \ \&\ 1_b \circ f = f. \ \square$$

3.1.2. DEFINITION. If $\mathcal{C}$ is a category and a and b are objects of $\mathcal{C}$, then

$$\mathcal{C}[a,b] = \{f \in M \mid f : a \to b\}. \ \square$$

In the following $\mathcal{C}$ is a category. The category $\mathcal{C}$ is assumed to be such that $\mathcal{C}[a,b]$ is a set (as opposed to a proper class) for all $a, b \in O$.

3.1.3. EXAMPLES. (i) **Set** is the category with sets as objects and functions as morphisms.

(ii) **Mon** is the category with complete partial orders (cpo's) as objects and monotone functions as morphisms. This category will be defined in section 3.4.

(iii) **Cpo** is the category with cpo's as objects and continuous functions as morphisms. $\square$

A functor is a morphism between categories, i.e. a structure-preserving map from one category into another.

3.1.4. DEFINITION. Let $\mathcal{C}$ and $\mathcal{D}$ be categories. A functor from $\mathcal{C}$ into $\mathcal{D}$ is a pair $\Phi = (\Phi_1, \Phi_2)$ of mappings

$$\Phi_1 : O(\mathcal{C}) \to O(\mathcal{D})$$

$$\Phi_2 : M(\mathcal{C}) \to M(\mathcal{D})$$

with the properties (i), (ii) and (iii) described below. For $a \in O(\mathcal{C})$ and $f \in M(\mathcal{C})$ we use the following notation:

$$\Phi_1(a) = \Phi(a) = \Phi a$$

$$\Phi_2(f) = \Phi(f) = \Phi f.$$

Now Φ is called a functor if it has the following three properties:

(i) $f : a \to b$ in $\mathcal{C} \Rightarrow \Phi f : \Phi a \to \Phi b$ in $\mathcal{D}$.

(ii) If $f : a \to b$ and $g : b \to c$, then

$$\Phi(g \circ f) = \Phi(g) \circ \Phi(f).$$

(iii) For every $a \in O(\mathcal{C})$

$$\Phi(1_a) = 1_{\Phi(a)}.$$

If Φ is a functor from $\mathcal{C}$ into $\mathcal{D}$, then we write $\Phi : \mathcal{C} \to \mathcal{D}$. $\square$

3.1.5. DEFINITION. Let $f : a \to b$ and $g : b \to c$.

(i) g is a left-inverse of f if $g \circ f = 1_a$.

(ii) If f has a left-inverse, then this is denoted by $f : a \rightarrowtail b$ and f is called left-invertible.

(iii) g is a right-inverse of f if $f \circ g = 1_b$.

(iv) If f has a right-inverse, then this is denoted by $f : a \twoheadrightarrow b$ and f is called right-invertible. □

If $f : a \to b$ has a left-inverse g and a right-inverse h, then

$$g = g \circ 1_b = g \circ (f \circ h) = (g \circ f) \circ h = 1_a \circ h = h$$

so $g = h$.

3.1.6. Definition. Let $a, b \in O$.

(i) If $f : a \to b$ is both left-invertible and right-invertible, then f is called invertible (notation $f : a \rightarrowtail\!\!\!\twoheadrightarrow b$). An invertible morphism is called an isomorphism.

(ii) If $f : a \rightarrowtail\!\!\!\twoheadrightarrow b$ is an isomorphism, then by the remark above, f has a unique inverse $f^{-1} : b \to a$ with the properties

$$f^{-1} \circ f = 1_a \;\&\; f \circ f^{-1} = 1_b.$$

(iii) The objects a and b are called isomorphic (in $\mathcal{C}$) if there exists an isomorphism $f : a \rightarrowtail\!\!\!\twoheadrightarrow b$. □

3.1.7. Definition. A terminal object is an object $\mathbf{1}$ such that for every object a there exists exactly one morphism $a \to \mathbf{1}$. This morphism is denoted by I_a. □

3.1.8. Examples. (i) Every singleton set is terminal in **Set**.

(ii) Every singleton cpo is terminal in **Mon**. □

Inspired by the fact that the elements of a set A correspond to functions from a singleton into A, points of objects are defined in the following way.

Let $\mathcal{C}$ be a category with terminal objects, and let $\mathbf{1}$ be a fixed terminal object in $\mathcal{C}$.

3.1.9. Definition. Let $a \in O$.

(i) A point of a is a morphism $p : \mathbf{1} \to a$.

(ii) $|a| = \mathcal{C}[\mathbf{1}, a]$ is the set of all points of a.

(iii) a is non-empty if $|a| \neq \emptyset$. □

3.1.10. Definition. Let $f : a \to b$. The function $|f| : |a| \to |b|$ is defined by

$$|f|(p) = f \circ p$$

for all $p \in |a|$. □

3.1.11. Lemma. *The mapping* $|\ |$ *from objects of* $\mathcal{C}$ *to sets and from morphisms of* $\mathcal{C}$ *to functions is a functor from* $\mathcal{C}$ *into* **Set**.

PROOF. (i) $f : a \to b$ in $\mathcal{C} \Rightarrow |f| : |a| \to |b|$ in **Set**.

(ii) Let $f : a \to b$ and $g : b \to c$. Then for all $p \in |a|$

$$\begin{aligned} |g \circ f|(p) &= (g \circ f) \circ p \\ &= g \circ (f \circ p) \\ &= |g|(f \circ p) \\ &= |g|(|f|(p)) \\ &= (|g| \circ |f|)(p) \end{aligned}$$

so $|g \circ f| = |g| \circ |f|$.

(iii) Let $a \in O$. Then for all $p \in |a|$

$$|1_a|(p) = 1_a \circ p = p = 1_{|a|}(p)$$

so $|1_a| = 1_{|a|}$. □

3.1.12. DEFINITION. (i) Extensional equality of morphisms is defined as follows. Let $f, g : a \to b$. Then

$$f \approx g \iff |f| = |g|.$$

(ii) The category $\mathcal{C}$ is extensional if for all $a, b \in O$ and all $f, g : a \to b$

$$f \approx g \Rightarrow f = g. \ \square$$

So $\mathcal{C}$ is extensional if and only if the functor $|\ |$ is faithful (i.e. injective on morphisms). The categories **Set**, **Mon** and **Cpo** are extensional.

3.1.13. DEFINITION. Let $a, b \in O$, $\varphi : |a| \to |b|$. Then φ is $\mathcal{C}$-representable if there exists $f : a \to b$ such that $\varphi = |f|$. If $f : a \to b$ is such that $|f| = \varphi$, then f is said to represent φ. □

3.1.14. DEFINITION. (i) Let $a, b \in O$. A product of a and b consists of an object $a \times b$ and two morphisms $\pi_1 : a \times b \to a$ and $\pi_2 : a \times b \to b$ such that for every $c \in O$ and any two morphisms $f : c \to a$ and $g : c \to b$ there exists a unique morphism $h : c \to a \times b$ such that

$$\pi_1 \circ h = f \ \& \ \pi_2 \circ h = g.$$

This unique h is denoted by $[f, g]$ and is called the pair of f and g. The morphisms π_1 and π_2 are called projections.

(ii) If any two objects in $\mathcal{C}$ have a product, then $\mathcal{C}$ is said to have products. □

From now on $\mathcal{C}$ is assumed to have products and a terminal object $\mathbf{1}$.

3.1.15. LEMMA. *For all $a, b, c \in O$, $f : c \to a$ and $g : c \to b$*

(i) $\pi_1 \circ [f, g] = f$.
(ii) $\pi_2 \circ [f, g] = g$.
(iii) *for every* $h : c \to a \times b$

$$\pi_1 \circ h = f \,\&\, \pi_2 \circ h = g \Rightarrow h = [f, g].$$

PROOF. By 3.1.14. □

3.1.16. LEMMA. (i) $[\pi_1, \pi_2] = 1_{a\times b}$.
(ii) $[f, g] \circ h = [f \circ h, g \circ h]$.

PROOF. By the previous lemma. □

3.1.17. DEFINITION. For all $f : a \to c$ and $g : b \to d$ define the morphism $f \times g : a \times b \to c \times d$ by

$$f \times g = [f \circ \pi_1, g \circ \pi_2].$$

The morphism $f \times g$ is called the product of f and g. □

3.1.18. LEMMA. (i) $1_a \times 1_b = 1_{a\times b}$.
(ii) $(f \times g) \circ [h, k] = [f \circ h, g \circ k]$.
(iii) $(f \times g) \circ (h \times k) = (f \circ h) \times (g \circ k)$.

PROOF. (i) $1_a \times 1_b = [1_a \circ \pi_1, 1_b \circ \pi_2] = [\pi_1, \pi_2] = 1_{a\times b}$.
(ii) Let $f : a \to b$, $g : c \to d$, $h : e \to a$ and $k : e \to b$. Then

$$\begin{aligned}(f \times g) \circ [h, k] &= [f \circ \pi_1, g \circ \pi_2] \circ [h, k] \\ &= [f \circ \pi_1 \circ [h, k], g \circ \pi_2 \circ [h, k]] \\ &= [f \circ h, g \circ h].\end{aligned}$$

(iii) Let $h : a_1 \to b_1$, $k : a_2 \to b_2$, $f : b_1 \to c_1$ and $g : b_2 \to c_2$. Then

$$\begin{aligned}(f \times g) \circ (h \times k) &= (f \times g) \circ [h \circ \pi_1, k \circ \pi_2] \\ &= [f \circ h \circ \pi_1, g \circ k \circ \pi_2] \\ &= (f \circ h) \times (g \circ k). \ \square\end{aligned}$$

3.1.19. LEMMA. (i) *If* $f : a \rightarrowtail\!\!\!\!\twoheadrightarrow c$ *and* $g : b \rightarrowtail\!\!\!\!\twoheadrightarrow d$, *then* $f \times g : a \times b \rightarrowtail\!\!\!\!\twoheadrightarrow c \times d$ *and* $(f \times g)^{-1} = f^{-1} \times g^{-1}$.
(ii) $a \cong c \,\&\, b \cong d \Rightarrow a \times b \cong c \times d$.

PROOF. (i) Let $f : a \rightarrowtail\!\!\!\!\twoheadrightarrow c$ and $g : b \rightarrowtail\!\!\!\!\twoheadrightarrow d$. Then by 3.1.18

$$(f^{-1} \times g^{-1}) \circ (f \times g) = 1_{a\times b}$$

$$(f \times g) \circ (f^{-1} \times g^{-1}) = 1_{c\times d}.$$

So $f \times g$ is invertible and $(f \times g)^{-1} = f^{-1} \times g^{-1}$.

(ii) By (i). □

These definitions and results on binary products can easily be generalized to arbitrary finite products. For $n \in \omega$ and $a_1, \dots a_n \in O$ define the product $a_1 \times \dots \times a_n$ by induction on n as follows.

- For $n = 0$: $a_1 \times \dots \times a_n = \mathbf{1}$.
- $a_1 \times \dots \times a_{n+1} = (a_1 \times \dots \times a_n) \times a_{n+1}$.

Then we have projections $\pi_i : a_1 \times \dots \times a_n \to a_i$ $(i = 1, \dots, n)$, finite sequences $[f_1, \dots, f_n]$ of morphisms, and finite products $f_1 \times \dots \times f_n$ of morphisms.

3.1.20. Definition. (i) Let $a, b \in O$. An exponent of a and b in $\mathcal{C}$ consists of an object b^a and a morphism $ev : b^a \times a \to b$ such that for every $c \in O$ and every $f : c \times a \to b$ there exists exactly one morphism $g : c \to b^a$ such that

$$ev \circ (g \times 1_a) = f.$$

This unique g is denoted by $\Lambda(f)$ or also by Λf. The morphism ev is called an evaluation morphism.

(ii) If $\mathcal{C}$ has an exponent for any two objects, then $\mathcal{C}$ is said to have exponentiation. □

3.1.21. Lemma. *Let $\mathcal{C}$ have exponentiation. Let $a, b \in O$ and let $f : c \times a \to b$. Then $\Lambda(f) : c \to b^a$ and*

$$ev \circ (\Lambda(f) \times 1_a) = f$$

and for every $g : c \to b^a$

$$ev \circ (g \times 1_a) = f \Rightarrow g = \Lambda(f).$$

Proof. By 3.1.20. □

3.1.22. Definition. A cartesian closed category, or a ccc, is a category that has products, a terminal object and exponentiation. □

3.1.23. Examples. (i) **Set** is a ccc.

(ii) **Mon** is a ccc. This will be shown in section 3.4.

(iii) **Cpo** is a ccc. □

In the following $\mathcal{C}$ is a cartesian closed category.

3.1.24. Lemma. *Let $a, b, c, d \in O$.*

(i) $\Lambda(ev) = 1_{b^a}$.

(ii) *If* $g : c \to b^a$, *then*
$$\Lambda(ev \circ (g \times 1_a)) = g.$$

(iii) *If* $f : c \times a \to b$ *and* $g : c \to a$, *then*
$$ev \circ [\Lambda(f), g] = f \circ [1_c, g].$$

(iv) *If* $f : c \times a \to b$ *and* $g : d \to a$, *then*
$$ev \circ (\Lambda(f) \times g) = f \circ (1_c \times g).$$

(v) *If* $f : c \times a \to b$ *and* $g : d \to c$, *then*
$$\Lambda(f) \circ g = \Lambda(f \circ (g \times 1_a)).$$

PROOF. (i) $ev \circ (1_{b^a} \times 1_a) = ev \circ 1_{b^a \times a} = ev$, so $\Lambda(ev) = 1_{b^a}$.

(ii) Let $g : c \to b^a$. Define $f = ev \circ (g \times 1_a)$. Then $g = \Lambda(f)$, so $\Lambda(ev \circ (g \times 1_a)) = g$.

(iii) Let $f : c \times a \to b$ and $g : c \to a$. Then
$$\begin{aligned} ev \circ [\Lambda(f), g] &= ev \circ [\Lambda(f) \circ 1_c, 1_a \times g] \\ &= ev \circ (\Lambda(f) \times 1_a) \circ [1_c, g] \\ &= f \circ [1_c, g]. \end{aligned}$$

(iv) Let $f : c \times a \to b$ and $g : d \to a$. Then
$$\begin{aligned} ev \circ (\Lambda(f) \times g) &= ev \circ ((\Lambda(f) \circ 1_c) \times (1_a \circ g)) \\ &= ev \circ (\Lambda(f) \times 1_a) \circ (1_c \times g) \\ &= f \circ (1_c \times g). \end{aligned}$$

(v) Let $f : c \times a \to b$ and $g : d \to c$. Then
$$\begin{aligned} ev \circ ((\Lambda(f) \circ g) \times 1_a) &= ev \circ ((\Lambda(f) \circ g) \times (1_a \circ 1_a)) \\ &= ev \circ (\Lambda(f) \times 1_a) \circ (g \times 1_a) \\ &= f \circ (g \times 1_a) \end{aligned}$$

so $\Lambda(f) \circ g = \Lambda(f \circ (g \times 1_a))$. $\square$

Using the Λ-operator we can define a weakening of extensionality for ccc's.

3.1.25. DEFINITION. (i) Λ is extensional if for all $a, b, c \in O$ and $f, g : c \times a \to b$
$$f \approx g \Rightarrow \Lambda(f) \approx \Lambda(g).$$

(ii) $\mathcal{C}$ is λ-extensional if Λ is extensional. $\square$

Clearly, every extensional ccc is λ-extensional.

Note that for all $a, b, c \in O$ the map Λ establishes a bijection from $\mathcal{C}[c \times a, b]$ onto $\mathcal{C}[c, b^a]$. Taking $c = \mathbf{1}$ we see that

$$\mathcal{C}[a, b] \sim \mathcal{C}[\mathbf{1} \times a, b] \sim \mathcal{C}[\mathbf{1}, b^a] = |b^a|$$

so morphisms $a \to b$ correspond to points of b^a. In the following we will make this a bit more precise.

3.1.26. Lemma. *Let $a \in O$. Then*

$$\pi_2 : \mathbf{1} \times a \rightarrowtail\!\!\!\!\rightarrow a$$

$$[I_a, 1_a] : a \rightarrowtail\!\!\!\!\rightarrow \mathbf{1} \times a$$

are each others inverses, and so $\mathbf{1} \times a \cong a$.

Proof. Clearly $\pi_2 \circ [I_a, 1_a] = 1_a$, and further

$$\begin{aligned} [I_a, 1_a] \circ \pi_2 &= [I_a \circ \pi_2, 1_a \circ \pi_2] \\ &= [I_{\mathbf{1} \times a}, \pi_2] \\ &= [\pi_1, \pi_2] \\ &= 1_{\mathbf{1} \times a}. \ \Box \end{aligned}$$

If $f : a \to b$, then $f \circ \pi_2 : \mathbf{1} \times a \to b$, so $\Lambda(f \circ \pi_2) : \mathbf{1} \to b^a$, so $\Lambda(f \circ \pi_2) \in |b^a|$.

3.1.27. Definition. For every $f : a \to b$ define

$$\ulcorner f \urcorner = \Lambda(f \circ \pi_2).$$

The point $\ulcorner f \urcorner$ of b^a is called the name of f. $\Box$

3.1.28. Lemma. *For all $f : a \to b$ and $p \in |a|$*

$$ev \circ [\ulcorner f \urcorner, p] = |f|(p).$$

Proof. Let $f : a \to b$ and $p \in |a|$. Then

$$\begin{aligned} ev \circ [\ulcorner f \urcorner, p] &= ev \circ [\Lambda(f \circ \pi_2), p] \\ &= f \circ \pi_2 \circ [1_{\mathbf{1}}, p] \\ &= f \circ p \\ &= |f|(p). \ \Box \end{aligned}$$

3.1.29. Lemma. *Let $a, b \in O$. Then $\ulcorner\ \urcorner$ gives a bijection from $\mathcal{C}[a, b]$ onto $|b^a|$.*

PROOF. For every point p of b^a we define a morphism $mor(p) : a \to b$ of which p is the name as follows.

$$mor(p) = ev \circ (p \times 1_a) \circ [I_a, 1_a].$$

$$\begin{tikzcd} a \arrow[r, "mor(p)"] \arrow[d, "{[I_a, 1_a]}"'] & b \\ \mathbf{1} \times a \arrow[r, "p \times 1_a"'] & b^a \times a \arrow[u, "ev"'] \end{tikzcd}$$

It will now be shown that mor and $\ulcorner\ \urcorner$ are each others inverses.

(i) Let $p \in |b^a|$. Then, using 3.1.26

$$\begin{aligned} \ulcorner mor(p) \urcorner &= \Lambda(mor(p) \circ \pi_2) \\ &= \Lambda(ev \circ (p \times 1_a) \circ [I_a, 1_a] \circ \pi_2) \\ &= \Lambda(ev \circ (p \times 1_a)) \\ &= p. \end{aligned}$$

(ii) Let $f : a \to b$. Then, using 3.1.26

$$\begin{aligned} mor(\ulcorner f \urcorner) &= ev \circ (\ulcorner f \urcorner \times 1_a) \circ [I_a, 1_a] \\ &= ev \circ (\Lambda(f \circ \pi_2) \times 1_a) \circ [I_a, 1_a] \\ &= f \circ \pi_2 \circ [I_a, 1_a] \\ &= f. \ \square \end{aligned}$$

Next it will be shown that for all objects a, b, c, d

$$a \cong c \,\&\, b \cong d \Rightarrow b^a \cong d^c.$$

We will in fact prove a stronger result that we are going to need in the next section.

3.1.30. THEOREM. *Let $f : a \rightarrowtail\!\!\!\!\rightarrow c$ and $g : b \rightarrowtail\!\!\!\!\rightarrow d$. Then there exists an isomorphism*

$$I(f, g) : b^a \rightarrowtail\!\!\!\!\rightarrow d^c$$

such that

$$ev \circ (I(f, g) \times f) = g \circ ev.$$

$$\begin{tikzcd} b^a \times a \arrow[r, "I(f,g) \times f"] \arrow[d, "ev"'] & d^c \times c \arrow[d, "ev"] \\ b \arrow[r, "g"'] & d \end{tikzcd}$$

Proof. We have $f : a \rightarrowtail\!\!\!\!\twoheadrightarrow c$, $f^{-1} : c \rightarrowtail\!\!\!\!\twoheadrightarrow a$, $g : b \rightarrowtail\!\!\!\!\twoheadrightarrow d$ and $g^{-1} : d \rightarrowtail\!\!\!\!\twoheadrightarrow b$. Define

$$h = g \circ ev \circ (1_{b^a} \times f^{-1}) : b^a \times c \to d$$

$$k = g^{-1} \circ ev \circ (1_{d^c} \times f) : d^c \times a \to b.$$

Then $\Lambda(h) : b^a \to d^c$ and $\Lambda(k) : d^c \to b^a$. It will be shown that $\Lambda(h)$ and $\Lambda(k)$ are each others inverses. First note that

$$ev \circ (\Lambda(h) \times f) = g \circ ev$$

$$ev \circ (\Lambda(k) \times f^{-1}) = g^{-1} \circ ev.$$

Therefore

$$\begin{aligned} ev \circ ((\Lambda(k) \circ \Lambda(h)) \times 1_a) &= ev \circ ((\Lambda(k) \circ \Lambda(h)) \times (f^{-1} \circ f)) \\ &= ev \circ (\Lambda(k) \times f^{-1}) \circ (\Lambda(h) \times f) \\ &= g^{-1} \circ ev \circ (\Lambda(h) \times f) \\ &= g^{-1} \circ g \circ ev \\ &= ev \end{aligned}$$

so $\Lambda(k) \circ \Lambda(h) = \Lambda(ev) = 1_{b^a}$. By symmetry we also have $\Lambda(h) \circ \Lambda(k) = 1_{d^c}$. Now define $I(f,g) = \Lambda(h)$. Then $I(f,g) : b^a \rightarrowtail\!\!\!\!\twoheadrightarrow d^c$ and $ev \circ (I(f,g) \times f) = g \circ ev$. $\square$

3.1.31. Corollary. $a \cong c \,\&\, b \cong d \Rightarrow b^a \cong d^c$. $\square$

3.2. Type structures in cartesian closed categories.

The notion of type structure will be generalized. For a ccc $\mathcal{C}$ we are going to define the notion of type structure in $\mathcal{C}$. An 'ordinary' type structure as defined in section 2.1 will then be a type structure in the ccc **Set**. The notions of homomorphism and isomorphism will be generalized as well.

Extensionality will also be generalized, but we have not been able to find a nice generalization of quasi-extensionality. Dual to extensionality is the notion of completeness. Type structures (in $\mathcal{C}$) that are both complete and extensional are called complex. In this section we will prove a representation theorem for the complex type structures in $\mathcal{C}$. This involves type structures $\mathbf{m}_{\mathcal{C}}(a)$ in $\mathcal{C}$ that are generated from non-empty objects a of $\mathcal{C}$. These type structures $\mathbf{m}_{\mathcal{C}}(a)$ are the generalizations of the full type structures $\mathfrak{M}(A)$ over non-empty sets.

Every type structure $\mathbf{m}$ in $\mathcal{C}$ gives rise to an ordinary type structure $\mathbf{m}^*$ in **Set**. So for every non-empty object a of $\mathcal{C}$ we obtain an ordinary type structure $\mathbf{m}_{\mathcal{C}}(a)^*$ which is denoted by $\mathfrak{M}_{\mathcal{C}}(a)$. In sections 3.3 and 3.5 these type structures $\mathfrak{M}_{\mathcal{C}}(a)$ will be used to construct algebras and models for simply typed λ-calculi.

Although the type structures $\mathfrak{M}_{\mathcal{C}}(a)$ have appeared in the literature before, the type structures $\mathbf{m}_{\mathcal{C}}(a)$ have appeared only implicitly, and the generalized type structures in cartesian closed categories seem not to have been studied before at all.

In the following $\mathcal{C}$ is a cartesian closed category.

3.2.1. DEFINITION. A type structure in $\mathcal{C}$ is a structure

$$\mathbf{m} = ((\mathbf{m}_\sigma)_{\sigma\in\mathbb{T}}, (app_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}})$$

such that for all $\sigma, \tau \in \mathbb{T}$

(i) $\mathbf{m}_\sigma$ is a non-empty object of $\mathcal{C}$

(ii) $app_{\sigma,\tau} : \mathbf{m}_{\sigma\to\tau} \times \mathbf{m}_\sigma \to \mathbf{m}_\tau$. $\square$

In the following $\mathbf{m}$, $\mathbf{m}^1$, $\mathbf{m}^2$, $\mathbf{m}^3, \ldots$ are type structures in $\mathcal{C}$, with

$$\mathbf{m} = ((\mathbf{m}_\sigma)_{\sigma\in\mathbb{T}}, (app_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}})$$

and for $i = 1, 2, 3, \ldots$

$$\mathbf{m}^i = ((\mathbf{m}^i_\sigma)_{\sigma\in\mathbb{T}}, (app^i_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}).$$

3.2.2. DEFINITION. A homomorphism from $\mathbf{m}^1$ to $\mathbf{m}^2$ is a family $f = (f_\sigma)_{\sigma\in\mathbb{T}}$ of morphisms $f_\sigma : \mathbf{m}^1_\sigma \to \mathbf{m}^2_\sigma$ such that for all $\sigma, \tau \in \mathbb{T}$

$$f_\tau \circ app^1_{\sigma,\tau} = app^2_{\sigma,\tau} \circ (f_{\sigma\to\tau} \times f_\sigma).$$

$$\begin{tikzcd}
\mathbf{m}^1_{\sigma\to\tau} \times \mathbf{m}^1_\sigma \arrow[r, "app^1_{\sigma,\tau}"] \arrow[d, "f_{\sigma\to\tau} \times f_\sigma"'] & \mathbf{m}^1_\tau \arrow[d, "f_\tau"] \\
\mathbf{m}^2_{\sigma\to\tau} \times \mathbf{m}^2_\sigma \arrow[r, "app^2_{\sigma,\tau}"'] & \mathbf{m}^2_\tau
\end{tikzcd}$$

If f is a homomorphism from $\mathbf{m}^1$ to $\mathbf{m}^2$, then this is denoted by $f : \mathbf{m}^1 \to \mathbf{m}^2$.
$\square$

Note that this notion of homomorphism is a generalization of the notion of homomorphism for ordinary type structures.

3.2.3. DEFINITION. (i) Let $f : \mathbf{m}^1 \to \mathbf{m}^2$ with $f = (f_\sigma)_{\sigma\in\mathbb{T}}$ and $g : \mathbf{m}^2 \to \mathbf{m}^3$ with $g = (g_\sigma)_{\sigma\in\mathbb{T}}$. The composition of f and g is defined by

$$g \circ f = (g_\sigma \circ f_\sigma)_{\sigma\in\mathbb{T}}.$$

(ii) The identity on $\mathbf{m}$ is defined by

$$1_{\mathbf{m}} = (1_{\mathbf{m}_\sigma})_{\sigma\in\mathbb{T}}. \ \square$$

3.2.4. LEMMA. (i) *If $f : \mathbf{m}^1 \to \mathbf{m}^2$ and $g : \mathbf{m}^2 \to \mathbf{m}^3$, then $g \circ f : \mathbf{m}^1 \to \mathbf{m}^3$.*
(ii) *$1_{\mathbf{m}} : \mathbf{m} \to \mathbf{m}$.*

PROOF. (i) Let $f : \mathbf{m}^1 \to \mathbf{m}^2$ and $g : \mathbf{m}^2 \to \mathbf{m}^3$. Define $h = g \circ f$. Then $h_\sigma = g_\sigma \circ f_\sigma$ for every $\sigma \in \mathbb{T}$ and for all $\sigma, \tau \in \mathbb{T}$

$$\begin{aligned}
h_\tau \circ app^1_{\sigma,\tau} &= (g_\tau \circ f_\tau) \circ app^1_{\sigma,\tau} \\
&= g_\tau \circ (f_\tau \circ app^1_{\sigma,\tau}) \\
&= g_\tau \circ (app^2_{\sigma,\tau} \circ (f_{\sigma\to\tau} \times f_\sigma)) \\
&= (g_\tau \circ app^2_{\sigma,\tau}) \circ (f_{\sigma\to\tau} \times f_\sigma) \\
&= (app^3_{\sigma,\tau} \circ (g_{\sigma\to\tau} \times g_\sigma)) \circ (f_{\sigma\to\tau} \times f_\sigma) \\
&= app^3_{\sigma,\tau} \circ ((g_{\sigma\to\tau} \times g_\sigma) \circ (f_{\sigma\to\tau} \times f_\sigma)) \\
&= app^3_{\sigma,\tau} \circ ((g_{\sigma\to\tau} \circ f_{\sigma\to\tau}) \times (g_\sigma \circ f_\sigma)) \\
&= app^3_{\sigma,\tau} \circ (h_{\sigma\to\tau} \times h_\sigma)
\end{aligned}$$

so $h : \mathbf{m}^1 \to \mathbf{m}^3$.
(ii) For all $\sigma, \tau \in \mathbb{T}$

$$\begin{aligned}
1_{\mathbf{m}_\tau} \circ app_{\sigma,\tau} &= app_{\sigma,\tau} \\
&= app_{\sigma,\tau} \circ 1_{(\mathbf{m}_{\sigma\to\tau} \times \mathbf{m}_\sigma)} \\
&= app_{\sigma,\tau} \circ (1_{\mathbf{m}_{\sigma\to\tau}} \times 1_{\mathbf{m}_\sigma})
\end{aligned}$$

so $1_{\mathbf{m}} : \mathbf{m} \to \mathbf{m}$. $\square$

3.2.5. PROPOSITION. *The type structures in $\mathcal{C}$ together with the homomorphisms form a category.*

PROOF. It is easy to check the Associativity Property and the Identity Property. $\square$

3.2.6. DEFINITION. (i) The category of type structures in $\mathcal{C}$ together with the homomorphisms is denoted by $\mathbf{TS}(\mathcal{C})$.
(ii) $\mathbf{TS} = \mathbf{TS}(\mathbf{Set})$. □

Next we will construct a functor * from $\mathbf{TS}(\mathcal{C})$ into $\mathbf{TS}$. This functor yields, for every type structure $\mathbf{m}$ in $\mathcal{C}$, an ordinary type structure $\mathbf{m}^*$. Remember the functor $|\ | : \mathcal{C} \to \mathbf{Set}$ that was defined in section 3.1.

3.2.7. DEFINITION. (i) $\mathbf{m}^*_\sigma = |\mathbf{m}_\sigma|$ for every $\sigma \in \mathbb{T}$.
(ii) Let $app^*_{\sigma,\tau} : \mathbf{m}^*_{\sigma\to\tau} \times \mathbf{m}^*_\sigma \to \mathbf{m}^*_\tau$ be defined by

$$app^*_{\sigma,\tau}(p,q) = app_{\sigma,\tau} \circ [p,q].$$

(iii) $\mathbf{m}^* = ((\mathbf{m}^*_\sigma)_{\sigma\in\mathbb{T}}, (app^*_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}})$.
(iv) Let $f : \mathbf{m}^1 \to \mathbf{m}^2$ with $f = (f_\sigma)_{\sigma\in\mathbb{T}}$. Define

$$f^* = (f^*_\sigma)_{\sigma\in\mathbb{T}}$$

with $f^*_\sigma = |f_\sigma|$ for every $\sigma \in \mathbb{T}$. □

3.2.8. PROPOSITION. ** is a functor from $\mathbf{TS}(\mathcal{C})$ into $\mathbf{TS}$.*

PROOF. (i) Let $f : \mathbf{m}^1 \to \mathbf{m}^2$ with $f = (f_\sigma)_{\sigma\in\mathbb{T}}$. We have to show that $f^* : (\mathbf{m}^1)^* \to (\mathbf{m}^2)^*$. Let $\sigma,\tau \in \mathbb{T}$, $p \in |\mathbf{m}^1_{\sigma\to\tau}|$, $q \in |\mathbf{m}^1_\sigma|$. Then

$$\begin{array}{rcl}
f^*_\tau(pq) & = & f^*_\tau(app^1_{\sigma,\tau} \circ [p,q]) \\
& = & |f_\tau|(app^1_{\sigma,\tau} \circ [p,q]) \\
& = & f_\tau \circ (app^1_{\sigma,\tau} \circ [p,q]) \\
& = & (f_\tau \circ app^1_{\sigma,\tau}) \circ [p,q] \\
& = & (app^2_{\sigma,\tau} \circ (f_{\sigma\to\tau} \times f_\sigma)) \circ [p,q] \\
& = & app^2_{\sigma,\tau} \circ ((f_{\sigma\to\tau} \times f_\sigma) \circ [p,q]) \\
& = & app^2_{\sigma,\tau} \circ [f_{\sigma\to\tau} \circ p, f_\sigma \circ q] \\
& = & app^2_{\sigma,\tau} \circ [f^*_{\sigma\to\tau}(p), f^*_\sigma(q)] \\
& = & f^*_{\sigma\to\tau}(p) f^*_\sigma(q)
\end{array}$$

so $f^* : (\mathbf{m}^1)^* \to (\mathbf{m}^2)^*$.
(ii) $(g \circ f)^* = g^* \circ f^*$.
(iii) $(1_{\mathbf{m}})^* = 1_{\mathbf{m}^*}$. □

Because $\mathbf{TS}(\mathcal{C})$ is a category we have the notions of isomorphism and of isomorphic type structures in $\mathcal{C}$.

3.2.9. LEMMA. *Let* $f : \mathbf{m}^1 \to \mathbf{m}^2$ *with* $f = (f_\sigma)_{\sigma\in\mathbb{T}}$. *Then* $f : \mathbf{m}^1 \rightarrowtail\!\!\!\rightarrow \mathbf{m}^2$ *if and only if for every* $\sigma \in \mathbb{T}$ *we have* $f_\sigma : \mathbf{m}^1_\sigma \rightarrowtail\!\!\!\rightarrow \mathbf{m}^2_\sigma$. *If* $f : \mathbf{m}^1 \rightarrowtail\!\!\!\rightarrow \mathbf{m}^2$, *then* $f^{-1} = (f_\sigma^{-1})_{\sigma\in\mathbb{T}}$.

PROOF. Simple. □

3.2.10. LEMMA. (i) $1_{\mathbf{m}} : \mathbf{m} \rightarrowtail\!\!\!\rightarrow \mathbf{m}$ *and* $(1_{\mathbf{m}})^{-1} = 1_{\mathbf{m}}$.
(ii) $f : \mathbf{m}^1 \rightarrowtail\!\!\!\rightarrow \mathbf{m}^2 \Rightarrow f^{-1} : \mathbf{m}^2 \rightarrowtail\!\!\!\rightarrow \mathbf{m}^1$.
(iii) $f : \mathbf{m}^1 \rightarrowtail\!\!\!\rightarrow \mathbf{m}^2 \,\&\, g : \mathbf{m}^2 \rightarrowtail\!\!\!\rightarrow \mathbf{m}^3 \Rightarrow g\circ f : \mathbf{m}^1 \rightarrowtail\!\!\!\rightarrow \mathbf{m}^3 \,\&\, (g\circ f)^{-1} = f^{-1}\circ g^{-1}$.

PROOF. Simple. □

We will now generalize the notion of extensionality to type structures in $\mathcal{C}$.

Let $\mathfrak{M}$ be an ordinary type structure. For all $\sigma, \tau \in \mathbb{T}$ and every $a \in \mathfrak{M}_{\sigma\to\tau}$ define the unary mapping $\hat{a} : \mathfrak{M}_\sigma \to \mathfrak{M}_\tau$ by

$$\hat{a}(b) = ab.$$

Note that $\mathfrak{M}$ is extensional if and only if the map $\hat{}$ is injective. The map $\hat{}$ can be defined categorically. For all $\sigma, \tau \in \mathbb{T}$ define

$$\Lambda_{\sigma,\tau} = \Lambda(App_{\sigma,\tau}).$$

Then $\Lambda_{\sigma,\tau}(a) = \hat{a}$ for every $a \in \mathfrak{M}_{\sigma\to\tau}$. So $\mathfrak{M}$ is extensional if and only if every $\Lambda_{\sigma,\tau}$ is injective, i.e. left-invertible.

Dually, $\mathfrak{M}$ is called complete if every $\Lambda_{\sigma,\tau}$ is surjective, i.e. right-invertible.

3.2.11. DEFINITION. (i) For all $\sigma, \tau \in \mathbb{T}$ define

$$\Lambda_{\sigma,\tau} = \Lambda(app_{\sigma,\tau}) : \mathbf{m}_{\sigma\to\tau} \to (\mathbf{m}_\tau)^{\mathbf{m}_\sigma}.$$

(ii) $\mathbf{m}$ is extensional if $\Lambda_{\sigma,\tau}$ is left-invertible for all $\sigma, \tau \in \mathbb{T}$.
(iii) $\mathbf{m}$ is complete if $\Lambda_{\sigma,\tau}$ is right-invertible for all $\sigma, \tau \in \mathbb{T}$.
(iv) $\mathbf{m}$ is complex if $\mathbf{m}$ is both complete and extensional. □

So the type structure $\mathbf{m}$ is complex if and only if every $\Lambda_{\sigma,\tau}$ is invertible.

NOTATION. $\Lambda_{\sigma,\tau}$ for $\mathbf{m}^i$ is denoted by $\Lambda^i_{\sigma,\tau}$. So for all i and all $\sigma, \tau \in \mathbb{T}$

$$\Lambda^i_{\sigma,\tau} = \Lambda(app^i_{\sigma,\tau}) : \mathbf{m}^i_{\sigma\to\tau} \to (\mathbf{m}^i_\tau)^{\mathbf{m}^i_\sigma}. \ \square$$

3.2.12. PROPOSITION. *If* $\mathcal{C}$ *is extensional and* $\mathbf{m}$ *is extensional, then* $\mathbf{m}^*$ *is extensional.*

PROOF. Let $\mathcal{C}$ be extensional and let $\mathbf{m}$ be extensional.

(i) First it will be shown that for all $\sigma, \tau \in \mathbb{T}$ and every $p \in \mathbf{m}^*_{\sigma\to\tau}$ there exists $f : \mathbf{m}_\sigma \to \mathbf{m}_\tau$ such that for all $q \in \mathbf{m}^*_\sigma$

$$pq = |f|(q).$$

Let $\sigma, \tau \in \mathbb{T}$ and let $p \in \mathbf{m}^*_{\sigma\to\tau}$. Then $\Lambda_{\sigma,\tau} \circ p : \mathbf{1} \to (\mathbf{m}_\tau)^{\mathbf{m}_\sigma}$, so there exists $f : \mathbf{m}_\sigma \to \mathbf{m}_\tau$ such that $\ulcorner f \urcorner = \Lambda_{\sigma,\tau} \circ p$. Then for all $q \in \mathbf{m}^*_\sigma$

$$\begin{aligned} pq &= app_{\sigma,\tau} \circ [p, q] \\ &= ev \circ (\Lambda_{\sigma,\tau} \times 1_{\mathbf{m}_\sigma}) \circ [p, q] \\ &= ev \circ [\Lambda_{\sigma,\tau} \circ p, q] \\ &= ev \circ [\ulcorner f \urcorner, q] \\ &= |f|(q). \end{aligned}$$

(ii) Let $\sigma, \tau \in \mathbb{T}$ and let $p_1, p_2 \in \mathbf{m}^*_{\sigma\to\tau}$ be such that $\forall q \in \mathbf{m}^*_\sigma [\, p_1 q = p_2 q \,]$. By (i) there exist $f_1 : \mathbf{m}_\sigma \to \mathbf{m}_\tau$ and $f_2 : \mathbf{m}_\sigma \to \mathbf{m}_\tau$ with $\ulcorner f_i \urcorner = \Lambda_{\sigma,\tau} \circ p_i$ for $i = 1, 2$. Then by (i) we have that for all $q \in \mathbf{m}^*_\sigma$

$$|f_1|(q) = p_1 q = p_2 q = |f_2|(q).$$

So $|f_1| = |f_2|$. But $\mathcal{C}$ is extensional, so $f_1 = f_2$, so $\Lambda_{\sigma,\tau} \circ p_1 = \Lambda_{\sigma,\tau} \circ p_2$. But $\mathbf{m}$ is extensional, so $\Lambda_{\sigma,\tau}$ is left-invertible and therefore $p_1 = p_2$. So $\mathbf{m}^*$ is extensional. $\Box$

Completeness of $\mathbf{m}$ says something about the $\mathbf{m}^*$-representablility of mappings on $\mathbf{m}^*$.

3.2.13. PROPOSITION. *Let $\mathbf{m}$ be complete. Let $\sigma, \tau \in \mathbb{T}$ and let $\varphi : \mathbf{m}^*_\sigma \to \mathbf{m}^*_\tau$. Then φ is $\mathbf{m}^*$-representable if and only if φ is $\mathcal{C}$-representable.*

PROOF. Let $R_{\sigma,\tau} : (\mathbf{m}_\tau)^{\mathbf{m}_\sigma} \to \mathbf{m}_{\sigma\to\tau}$ be a right-inverse of $\Lambda_{\sigma,\tau}$.

(i) Let φ be $\mathbf{m}^*$-representable. Let $p \in \mathbf{m}^*_{\sigma\to\tau}$ $\mathbf{m}^*$-represent φ. Let $f : \mathbf{m}_\sigma \to \mathbf{m}_\tau$ be such that $\ulcorner f \urcorner = \Lambda_{\sigma,\tau} \circ p$. Then for all $q \in \mathbf{m}^*_\sigma$

$$\begin{aligned} \varphi(q) &= pq \\ &= app_{\sigma,\tau} \circ [p, q] \\ &= ev \circ (\Lambda_{\sigma,\tau} \times 1_{\mathbf{m}_\sigma}) \circ [p, q] \\ &= ev \circ [\Lambda_{\sigma,\tau} \circ p, q] \\ &= ev \circ [\ulcorner f \urcorner, q] \\ &= |f|(q) \end{aligned}$$

so $\varphi = |f|$, and so φ is $\mathcal{C}$-representable.

(ii) Now suppose φ is $\mathcal{C}$-representable. Let $f : \mathbf{m}_\sigma \to \mathbf{m}_\tau$ be such that $\varphi = |f|$. Then $\ulcorner f \urcorner : \mathbf{1} \to (\mathbf{m}_\tau)^{\mathbf{m}_\sigma}$. Define $p = R_{\sigma,\tau} \circ \ulcorner f \urcorner : \mathbf{1} \to \mathbf{m}_{\sigma\to\tau}$, so $p \in \mathbf{m}^*_{\sigma\to\tau}$ and

$$\Lambda_{\sigma,\tau} \circ p = \Lambda_{\sigma,\tau} \circ R_{\sigma,\tau} \circ \ulcorner f \urcorner = \ulcorner f \urcorner.$$

So for all $q \in \mathbf{m}^*_\sigma$

$$\begin{aligned} \varphi(q) &= |f|(q) \\ &= ev \circ [\ulcorner f \urcorner, q] \\ &= ev \circ [\Lambda_{\sigma,\tau} \circ p, q] \\ &= ev \circ (\Lambda_{\sigma,\tau} \times 1_{\mathbf{m}_\sigma}) \circ [p, q] \\ &= app_{\sigma,\tau} \circ [p, q] \\ &= pq \end{aligned}$$

so φ is $\mathbf{m}^*$-representable. □

We will now define, for every non-empty object a of $\mathcal{C}$, a complex type structure $\mathbf{m}_\mathcal{C}(a)$. These type structures are the generalizations of the full type structures.

3.2.14. Definition. Let $a \in O$ with $|a| \neq \emptyset$.

(i) For every $\sigma \in \mathbb{T}$ define $a_\sigma \in O$ by induction on the structure of σ as follows.

(1) $a_0 = a$.

(2) $a_{\sigma\to\tau} = (a_\tau)^{a_\sigma}$.

(ii) For all $\sigma, \tau \in \mathbb{T}$ let $app_{\sigma,\tau} : a_{\sigma\to\tau} \times a_\sigma \to a_\tau$ be the evaluation morphism $ev : (a_\tau)^{a_\sigma} \times a_\sigma \to a_\tau$.

(iii) The type structure in $\mathcal{C}$ that is generated from a is defined by

$$\mathbf{m}_\mathcal{C}(a) = ((a_\sigma)_{\sigma\in\mathbb{T}}, (app_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}). \ \square$$

It can easily be shown that every a_σ is non-empty, so that $\mathbf{m}_\mathcal{C}(a)$ is indeed a type structure.

3.2.15. Proposition. *Let $a \in O$ be non-empty. Then $\mathbf{m}_\mathcal{C}(a)$ is complex.*

Proof. For all $\sigma, \tau \in \mathbb{T}$ we have that $\Lambda_{\sigma,\tau} = \Lambda(ev) = 1_{a_{\sigma\to\tau}}$, so $\Lambda_{\sigma,\tau}$ is invertible and therefore $\mathbf{m}_\mathcal{C}(a)$ is complex. □

3.2.16. Definition. For every non-empty object a of $\mathcal{C}$ define

$$\mathfrak{M}_\mathcal{C}(a) = \mathbf{m}_\mathcal{C}(a)^*. \ \square$$

3.2.17. LEMMA. *If $\mathcal{C}$ is extensional, then $\mathfrak{M}_{\mathcal{C}}(a)$ is extensional for every non-empty object a of $\mathcal{C}$.*

PROOF. By 3.2.12 and 3.2.15. □

In the following it will be shown that the type structures $\mathbf{m}_{\mathcal{C}}(a)$ are the only ones that are complex, modulo isomorphy. The main tool for this is the following Isomorphy Theorem.

3.2.18. THEOREM (Isomorphy Theorem).
If $\mathbf{m}^1$ and $\mathbf{m}^2$ are complex and $\mathbf{m}^1_0$ and $\mathbf{m}^2_0$ are isomorphic objects of $\mathcal{C}$, then $\mathbf{m}^1 \cong \mathbf{m}^2$.

PROOF. Suppose that $\mathbf{m}^1$ and $\mathbf{m}^2$ are complex and that $\mathbf{m}^1_0 \cong \mathbf{m}^2_0$. For every $\sigma \in \mathbb{T}$ define $f_\sigma : \mathbf{m}^1_\sigma \to \mathbf{m}^2_\sigma$ by induction on the structure of σ as follows.

(i) Let $f_0 : \mathbf{m}^1_0 \rightarrowtail\!\!\!\!\!\rightarrow \mathbf{m}^2_0$ be any isomorphism.

(ii) Let $f_\sigma : \mathbf{m}^1_\sigma \rightarrowtail\!\!\!\!\!\rightarrow \mathbf{m}^2_\sigma$ and $f_\tau : \mathbf{m}^1_\tau \rightarrowtail\!\!\!\!\!\rightarrow \mathbf{m}^2_\tau$. By theorem 3.1.30 we have an isomorphism

$$I(f_\sigma, f_\tau) : (\mathbf{m}^1_\tau)^{\mathbf{m}^1_\sigma} \rightarrowtail\!\!\!\!\!\rightarrow (\mathbf{m}^2_\tau)^{\mathbf{m}^2_\sigma}.$$

Define $f_{\sigma\to\tau} = (\Lambda^2_{\sigma,\tau})^{-1} \circ I(f_\sigma, f_\tau) \circ \Lambda^1_{\sigma,\tau}$. Then $f_{\sigma\to\tau} : \mathbf{m}^1_{\sigma\to\tau} \rightarrowtail\!\!\!\!\!\rightarrow \mathbf{m}^2_{\sigma\to\tau}$.
Now we have an isomorphism $f_\sigma : \mathbf{m}^1_\sigma \rightarrowtail\!\!\!\!\!\rightarrow \mathbf{m}^2_\sigma$ for every $\sigma \in \mathbb{T}$. Define $f = (f_\sigma)_{\sigma\in\mathbb{T}}$. All we need to prove now is that f is a homomorphism from $\mathbf{m}^1$ into $\mathbf{m}^2$. So let $\sigma, \tau \in \mathbb{T}$. Then we have the following equations:

- $app^1_{\sigma,\tau} = ev \circ (\Lambda^1_{\sigma,\tau} \times 1_{(\mathbf{m}^1_\sigma)})$.
- $f_\tau \circ ev = ev \circ (I(f_\sigma, f_\tau) \times f_\sigma)$.
- $ev = app^2_{\sigma,\tau} \circ ((\Lambda^2_{\sigma,\tau})^{-1} \times 1_{(\mathbf{m}^2_\sigma)})$.

Using these equations we compute as follows:

$$\begin{aligned}
f_\tau \circ app^1_{\sigma,\tau} &= f_\tau \circ ev \circ (\Lambda^1_{\sigma,\tau} \times 1_{(\mathbf{m}^1_\sigma)}) \\
&= ev \circ (I(f_\sigma, f_\tau) \times f_\sigma) \circ (\Lambda^1_{\sigma,\tau} \times 1_{(\mathbf{m}^1_\sigma)}) \\
&= app^2_{\sigma,\tau} \circ ((\Lambda^2_{\sigma,\tau})^{-1} \times 1_{(\mathbf{m}^2_\sigma)}) \circ (I(f_\sigma, f_\tau) \times f_\sigma) \circ (\Lambda^1_{\sigma,\tau} \times 1_{(\mathbf{m}^1_\sigma)}) \\
&= app^2_{\sigma,\tau} \circ (((\Lambda^2_{\sigma,\tau})^{-1} \circ I(f_\sigma, f_\tau) \circ \Lambda^1_{\sigma,\tau}) \times (1_{(\mathbf{m}^2_\sigma)} \circ f_\sigma \circ 1_{(\mathbf{m}^1_\sigma)})) \\
&= app^2_{\sigma,\tau} \circ (f_{\sigma\to\tau} \times f_\sigma).
\end{aligned}$$

So f is a homomorphism from $\mathbf{m}^1$ into $\mathbf{m}^2$. Therefore f is an isomorphism and so $\mathbf{m}^1 \cong \mathbf{m}^2$. □

3.2.19. THEOREM (Representation Theorem).
If $\mathbf{m}$ is a complex type structure in $\mathcal{C}$, then $\mathbf{m} \cong \mathbf{m}_{\mathcal{C}}(\mathbf{m}_0)$.

PROOF. Immediately from the Isomorphy Theorem. □

3.3. Algebras and models in cartesian closed categories.

Let $\mathcal{C}$ be a cartesian closed category. Let a be a non-empty object of $\mathcal{C}$. Let $\mathbf{m} = \mathbf{m}_{\mathcal{C}}(a)$ and $\mathfrak{M} = \mathbf{m}^* = \mathbf{m}_{\mathcal{C}}(a)^*$. Then $\mathbf{m}_0 = a$, $\mathbf{m}_{\sigma\to\tau} = (\mathbf{m}_\tau)^{\mathbf{m}_\sigma}$ and $\mathfrak{M}_\sigma = |\mathbf{m}_\sigma|$.

3.3.1. Definition. For every $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0 \in \mathbb{T}$ define

$$(\mathbf{m})_\sigma = \mathbf{1} \times \mathbf{m}_{\sigma_1} \times \ldots \times \mathbf{m}_{\sigma_k}. \ \square$$

In the following we will be using finite sequences of different variables. We will use Δ, Γ and B for these sequences.

3.3.2. Definition. Let $\Delta = (x_1, \ldots, x_k)$ and $\Gamma = (y_1, \ldots, y_l)$ be sequences of different variables.

(i) $x \in \Delta \Leftrightarrow x \in \{x_1, \ldots, x_k\}$.

(ii) $\Gamma \subseteq \Delta \Leftrightarrow \{y_1, \ldots, y_l\} \subseteq \{x_1, \ldots, x_k\}$. $\square$

3.3.3. Definition. Let $x_1, \ldots, x_k$ be different variables, with $x_i \in Var_{\sigma_i}$ $(i = 1, \ldots, k)$, $\Delta = (x_1, \ldots, x_k)$, $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0$.

(i) $(\mathbf{m})_\Delta = (\mathbf{m})_\sigma$.

(ii) $1_\Delta = 1_{(\mathbf{m})_\Delta}$.

(iii) $I_\Delta = I_{(\mathbf{m})_\Delta}$.

(iv) If $x \equiv x_i$, then π^Δ_x is the projection $(\mathbf{m})_\Delta \to \mathbf{m}_{\sigma_i}$. $\square$

3.3.4. Definition. Let $\Delta = (x_1, \ldots, x_k)$ and $\Gamma = (y_1, \ldots, y_l)$ be sequences of different variables, with $\Gamma \subseteq \Delta$. Define $\Pi^\Delta_\Gamma : (\mathbf{m})_\Delta \to (\mathbf{m})_\Gamma$ by

$$\Pi^\Delta_\Gamma = [I_\Delta, \pi^\Delta_{y_1}, \ldots, \pi^\Delta_{y_l}]. \ \square$$

3.3.5. Lemma. (i) $x \in \Gamma \subseteq \Delta \Rightarrow \pi^\Gamma_x \circ \Pi^\Delta_\Gamma = \pi^\Delta_x$.

(ii) $B \subseteq \Gamma \subseteq \Delta \Rightarrow \Pi^\Gamma \circ \Pi^\Delta_B = \Pi^\Delta_B$.

(iii) $\Pi^\Delta_\Delta = 1_\Delta$.

(iv) $\Gamma \subseteq \Delta \Rightarrow [\Pi^{\Delta,x}_\Gamma, \pi^{\Delta,x}_x] = \Pi^{\Delta,x}_{\Gamma,x}$.

(v) $[\Pi^{\Delta,x}_\Delta, \pi^{\Delta,x}_x] = 1_{\Delta,x}$.

(vi) $\Gamma \subseteq \Delta \;\&\; x \in Var_\sigma \Rightarrow \Pi^\Delta_\Gamma \times 1_{\mathbf{m}_\sigma} = \Pi^{\Delta,x}_{\Gamma,x}$.

Proof. The proofs of (i)-(v) are simple. The proof of (vi) is as follows. Let $\Gamma \subseteq \Delta$ and $x \in Var_\sigma$. Then

$$\begin{aligned}
\Pi^\Delta_\Gamma \times 1_{\mathbf{m}_\sigma} &= (\Pi^\Delta_\Gamma \times 1_{\mathbf{m}_\sigma}) \circ 1_{\Delta,x} \\
&= (\Pi^\Delta_\Gamma \times 1_{\mathbf{m}_\sigma}) \circ [\Pi^{\Delta,x}_\Delta, \pi^{\Delta,x}_x] \\
&= [\Pi^\Delta_\Gamma \circ \Pi^{\Delta,x}_\Delta, 1_{\mathbf{m}_\sigma} \circ \pi^{\Delta,x}_x] \\
&= [\Pi^{\Delta,x}_\Gamma, \pi^{\Delta,x}_x] \\
&= \Pi^{\Delta,x}_{\Gamma,x}. \ \square
\end{aligned}$$

If $x \in Var_\sigma$, then $(\mathbf{m})_{\Delta,x} = (\mathbf{m})_\Delta \times \mathbf{m}_\sigma$, and $[\Pi^{\Delta,x}_\Delta, \pi^{\Delta,x}_x] = 1_{\Delta,x}$ so $\Pi^{\Delta,x}_\Delta$ is the first projection $(\mathbf{m})_{\Delta,x} \to (\mathbf{m})_\Delta$ and $\pi^{\Delta,x}_x$ is the second projection $(\mathbf{m})_{\Delta,x} \to \mathbf{m}_\sigma$.

3.3.6. DEFINITION. Let $\Delta = (x_1, \ldots, x_k)$ be a sequence of different variables and let $\sigma \in \mathbb{T}$.

(i) $\mathcal{T}_\sigma(\mathbb{C})^\Delta = \{M \in \mathcal{T}_\sigma(\mathbb{C}) | \; FV(M) \subseteq \{x_1, \ldots, x_k\}\,\}$.

(ii) $\mathcal{T}(\mathbb{C})^\Delta = \{M \in \mathcal{T}(\mathbb{C}) | \; FV(M) \subseteq \{x_1, \ldots, x_k\}\,\}$. □

Let $\mu : |\mathbb{C}| \to |\mathfrak{M}|$ be type-preserving. Then for all $\sigma \in \mathbb{T}$ and $\mathbf{C} \in \mathbb{C}_\sigma$ we have

$$\mu(\mathbf{C}) : \mathbf{1} \to \mathbf{m}_\sigma.$$

3.3.7. DEFINITION. For every Δ, every $\sigma \in \mathbb{T}$ and every $M \in \mathcal{T}_\sigma(\mathbb{C})^\Delta$ we define a morphism

$$[\![M]\!]_\Delta : (\mathbf{m})_\Delta \to \mathbf{m}_\sigma$$

by induction on the structure of M, as follows.

(i) $[\![\mathbf{C}]\!]_\Delta = \mu(\mathbf{C}) \circ I_\Delta$.

(ii) $[\![x]\!]_\Delta = \pi^\Delta_x$.

(iii) $[\![MN]\!]_\Delta = ev \circ [[\![M]\!]_\Delta, [\![N]\!]_\Delta]$.

(iv) $[\![\lambda x.M]\!]_\Delta = \Lambda([\![M]\!]_{\Delta,x})$. □

3.3.8. LEMMA. *If $M \in \mathcal{T}(\mathbb{C})^\Gamma$ and $\Gamma \subseteq \Delta$, then*

$$[\![M]\!]_\Gamma \circ \Pi^\Delta_\Gamma = [\![M]\!]_\Delta.$$

PROOF. By induction on the structure of M.

(i) $[\![x]\!]_\Gamma \circ \Pi^\Delta_\Gamma = \pi^\Gamma_x \circ \Pi^\Delta_\Gamma = \pi^\Delta_x = [\![x]\!]_\Delta$.

(ii)

$$\begin{aligned}
[\![\mathbf{C}]\!]_\Gamma \circ \Pi^\Delta_\Gamma &= \mu(\mathbf{C}) \circ I_\Gamma \circ \Pi^\Delta_\Gamma \\
&= \mu(\mathbf{C}) \circ I_\Delta \\
&= [\![\mathbf{C}]\!]_\Delta.
\end{aligned}$$

(iii)

$$\begin{aligned}
[\![MN]\!]_\Gamma \circ \Pi^\Delta_\Gamma &= ev \circ [[\![M]\!]_\Gamma, [\![N]\!]_\Gamma] \circ \Pi^\Delta_\Gamma \\
&= ev \circ [[\![M]\!]_\Gamma \circ \Pi^\Delta_\Gamma, [\![N]\!]_\Gamma \circ \Pi^\Delta_\Gamma] \\
&= ev \circ [[\![M]\!]_\Delta, [\![N]\!]_\Delta] \\
&= [\![MN]\!]_\Delta.
\end{aligned}$$

(iv)

$$\begin{aligned} [\![\lambda x.M]\!]_\Gamma \circ \Pi_\Gamma^\Delta &= \Lambda([\![M]\!]_{\Gamma,x}) \circ \Pi_\Gamma^\Delta \\ &= \Lambda([\![M]\!]_{\Gamma,x} \circ (\Pi_\Gamma^\Delta \times 1_{\mathbf{m}_\sigma})) \\ &= \Lambda([\![M]\!]_{\Gamma,x} \circ \Pi_{\Gamma,x}^{\Delta,x}) \\ &= \Lambda([\![M]\!]_{\Delta,x}) \\ &= [\![\lambda x.M]\!]_\Delta. \ \Box \end{aligned}$$

3.3.9. Lemma. *Let* $M \in \mathcal{T}(\mathbb{C})^\Gamma$, $\Gamma = (x_1, \ldots, x_k)$, $\vec{N} = N_1, \ldots, N_k \in \mathcal{T}(\mathbb{C})^\Delta$ *be such that* $\vec{N}$ *fits in* $\vec{x} = x_1, \ldots, x_k$. *Then*

$$[\![M[\vec{x} := \vec{N}]]\!]_\Delta = [\![M]\!]_\Gamma \circ [I_\Delta, [\![\vec{N}]\!]_\Delta].$$

Proof. By induction on the structure of M.

(i) Let $x \equiv x_i$. Then

$$\begin{aligned} [\![x[\vec{x} := \vec{N}]]\!]_\Delta &= [\![N_i]\!]_\Delta \\ &= \pi_x^\Gamma \circ [I_\Delta, [\![\vec{N}]\!]_\Delta] \\ &= [\![x]\!]_\Gamma \circ [I_\Delta, [\![\vec{N}]\!]_\Delta]. \end{aligned}$$

(ii)

$$\begin{aligned} [\![\mathbf{C}[\vec{x} := \vec{N}]]\!]_\Delta &= [\![\mathbf{C}]\!]_\Delta \\ &= \mu(\mathbf{C}) \circ I_\Delta \\ &= \mu(\mathbf{C}) \circ I_\Gamma \circ [I_\Delta, [\![\vec{N}]\!]_\Delta]. \end{aligned}$$

(iii)

$$\begin{aligned} [\![(M_1 M_2)[\vec{x} := \vec{N}]]\!]_\Delta &= [\![M_1[\vec{x} := \vec{N}] M_2[\vec{x} := \vec{N}]]\!]_\Delta \\ &= ev \circ [[\![M_1[\vec{x} := \vec{N}]]\!]_\Delta, [\![M_2[\vec{x} := \vec{N}]]\!]_\Delta] \\ &= ev \circ [[\![M_1]\!]_\Gamma \circ [I_\Delta, [\![\vec{N}]\!]_\Delta], [\![M_2]\!]_\Gamma \circ [I_\Delta, [\![\vec{N}]\!]]] \\ &= ev \circ [[\![M_1]\!]_\Gamma, [\![M_2]\!]_\Gamma] \circ [I_\Delta, [\![\vec{N}]\!]_\Delta] \\ &= [\![M_1 M_2]\!]_\Gamma \circ [I_\Delta, [\![\vec{N}]\!]_\Delta]. \end{aligned}$$

(iv)

$$\begin{aligned} [\![(\lambda x.M)[\vec{x} := \vec{N}]]\!]_\Delta &= [\![\lambda x.M[\vec{x} := \vec{N}]]\!]_\Delta \\ &= \Lambda([\![M[\vec{x} := \vec{N}]]\!]_{\Delta,x}) \\ &= \Lambda([\![M[\vec{x} := \vec{N}, x := x]]\!]_{\Delta,x}) \\ &= \Lambda([\![M]\!]_{\Gamma,x} \circ [I_{\Delta,x}, [\![\vec{N}]\!]_{\Delta,x}, [\![x]\!]_{\Delta,x}]) \end{aligned}$$

$$\begin{aligned}
&= \Lambda(\llbracket M\rrbracket_{\Gamma,x}\circ[I_\Delta\circ\Pi^{\Delta,x}_\Delta, \llbracket\vec{N}\rrbracket_\Delta\circ\Pi^{\Delta,x}_\Delta, \pi^{\Delta,x}_x])\\
&= \Lambda(\llbracket M\rrbracket_{\Gamma,x}\circ[[I_\Delta, \llbracket\vec{N}\rrbracket_\Delta]\circ\Pi^{\Delta,x}_\Delta, 1_{\mathbf{m}_\sigma}\circ\pi^{\Delta,x}_x])\\
&= \Lambda(\llbracket M\rrbracket_{\Gamma,x}\circ([I_\Delta, \llbracket\vec{N}\rrbracket_\Delta]\times 1_{\mathbf{m}_\sigma}))\\
&= \Lambda(\llbracket M\rrbracket_{\Gamma,x})\circ[I_\Delta, \llbracket\vec{N}\rrbracket_\Delta]\\
&= \llbracket\lambda x.M\rrbracket_\Gamma\circ[I_\Delta, \llbracket\vec{N}\rrbracket_\Delta].\ \square
\end{aligned}$$

3.3.10. LEMMA. *Let $\Gamma\subseteq\Delta$, $\sigma\in\mathbb{T}$, $x\in Var_\sigma$, $M\in\mathcal{T}(\mathbb{C})^{\Gamma,x}$ and $N\in\mathcal{T}_\sigma(\mathbb{C})^\Delta$. Then*

$$\llbracket M[x:=N]\rrbracket_\Delta = \llbracket M\rrbracket_{\Gamma,x}\circ[\Pi^\Delta_\Gamma, \llbracket N\rrbracket_\Delta].$$

PROOF. Say $\Gamma=(x_1,\ldots,x_k)$. Then

$$\begin{aligned}
\llbracket M\rrbracket_{\Gamma,x}\circ[\Pi^\Delta_\Gamma, \llbracket N\rrbracket_\Delta] &= \llbracket M\rrbracket_{\Gamma,x}\circ[I_\Delta, \pi^\Delta_{x_1},\ldots,\pi^\Delta_{x_k}, \llbracket N\rrbracket_\Delta]\\
&= \llbracket M\rrbracket_{\Gamma,x}\circ[I_\Delta, \llbracket\vec{x}\rrbracket_\Delta, \llbracket N\rrbracket_\Delta]\\
&= \llbracket M[\vec{x}:=\vec{x}, x:=N]\rrbracket_\Delta\\
&= \llbracket M[x:=N]\rrbracket_\Delta.\ \square
\end{aligned}$$

Next we will define satisfaction of equations in the structure $(\mathbf{m}_\mathcal{C}(a), \llbracket\,\rrbracket)$.

3.3.11. DEFINITION. (i) Satisfaction in $(\mathbf{m}_\mathcal{C}(a), \llbracket\,\rrbracket)$ is defined as follows. For all $\sigma\in\mathbb{T}$, $M,N\in\mathcal{T}_\sigma(\mathbb{C})$

$$\mathbf{m}_\mathcal{C}(a), \llbracket\,\rrbracket \models M=N$$

if for every Δ with $M,N\in\mathcal{T}(\mathbb{C})^\Delta$ we have

$$\llbracket M\rrbracket_\Delta = \llbracket N\rrbracket_\Delta.$$

(ii) The set of all theorems of $(\mathbf{m}_\mathcal{C}(a), \llbracket\,\rrbracket)$ is defined by

$$Th(\mathbf{m}_\mathcal{C}(a), \llbracket\,\rrbracket) = \{(M=N)\mid \mathbf{m}_\mathcal{C}(a), \llbracket\,\rrbracket\models M=N\}.\ \square$$

3.3.12. PROPOSITION. (i) $\mathbf{m}_\mathcal{C}(a), \llbracket\,\rrbracket \models (\lambda x.M)N = M[x:=N]$.
(ii) $\mathbf{m}_\mathcal{C}(a), \llbracket\,\rrbracket \models \lambda x.Mx = M$ *if $x\notin FV(M)$.*

PROOF. (i) Let Δ be such that $((\lambda x.M)N)\in\mathcal{T}(\mathbb{C})^\Delta$. Then also $M[x:=N]\in\mathcal{T}(\mathbb{C})^\Delta$. And then

$$\begin{aligned}
\llbracket(\lambda x.M)N\rrbracket_\Delta &= ev\circ[\llbracket\lambda x.M\rrbracket_\Delta, \llbracket N\rrbracket_\Delta]\\
&= ev\circ[\Lambda(\llbracket M\rrbracket_{\Delta,x}), \llbracket N\rrbracket_\Delta]\\
&= \llbracket M\rrbracket_{\Delta,x}\circ[1_\Delta, \llbracket N\rrbracket_\Delta]\\
&= \llbracket M\rrbracket_{\Delta,x}\circ[\Pi^\Delta_\Delta, \llbracket N\rrbracket_\Delta]\\
&= \llbracket M[x:=N]\rrbracket_\Delta.
\end{aligned}$$

(ii) Let $x \notin FV(M)$. Let Δ be such that $M \in \mathcal{T}(\mathbb{C})^\Delta$. Then $(\lambda x.Mx) \in \mathcal{T}(\mathbb{C})^\Delta$ and

$$\begin{aligned}
[\![\lambda x.Mx]\!]_\Delta &= \Lambda([\![Mx]\!]_{\Delta,x}) \\
&= \Lambda(ev \circ [[\![M]\!]_{\Delta,x}, [\![x]\!]_{\Delta,x}]) \\
&= \Lambda(ev \circ [[\![M]\!]_\Delta \circ \Pi^{\Delta,x}_\Delta, \pi^{\Delta,x}_x]) \\
&= \Lambda(ev \circ [[\![M]\!]_\Delta \circ \Pi^{\Delta,x}_\Delta, 1_{\mathbf{m}_\sigma} \circ \pi^{\Delta,x}_x]) \\
&= \Lambda(ev \circ ([\![M]\!]_\Delta \times 1_{\mathbf{m}_\sigma})) \\
&= [\![M]\!]_\Delta. \ \Box
\end{aligned}$$

3.3.13. THEOREM. *$Th(\mathbf{m}_\mathcal{C}(a), [\![\ ]\!])$ is a $\lambda^\rightarrow\eta(\mathbb{C})$-theory.*

PROOF. By 1.2.25 it follows easily that $Th(\mathbf{m}_\mathcal{C}(a), [\![\ ]\!])$ is a theory. By the previous proposition we have $Ax(\lambda^\rightarrow\eta(\mathbb{C})) \subseteq Th(\mathbf{m}_\mathcal{C}(a), [\![\ ]\!])$. So $Th(\mathbf{m}_\mathcal{C}(a), [\![\ ]\!])$ is a $\lambda^\rightarrow\eta(\mathbb{C})$-theory. $\Box$

3.3.14. COROLLARY. $\lambda^\rightarrow\eta(\mathbb{C}) \vdash M = N \Rightarrow [\![M]\!]_\Delta = [\![N]\!]_\Delta$. $\Box$

3.3.15. DEFINITION. Let $\Delta = (x_1, \ldots, x_k)$ be a sequence of different variables. Let $\rho \in Val(\mathfrak{M}_\mathcal{C}(a))$. Then define $\rho^\Delta : \mathbf{1} \to (\mathbf{m})_\Delta$ by

$$\rho^\Delta = [1_\mathbf{1}, \rho(x_1), \ldots, \rho(x_k)]. \ \Box$$

3.3.16. LEMMA. (i) $\pi^\Delta_x \circ \rho^\Delta = \rho(x)$.
(ii) $\Pi^\Delta_\Gamma \circ \rho^\Delta = \rho^\Gamma$.

PROOF. (i) Simple.
(ii) Let $\Gamma = (y_1, \ldots, y_l)$. Then

$$\begin{aligned}
\Pi^\Delta_\Gamma \circ \rho^\Delta &= [I_\Delta, \pi^\Delta_{y_1}, \ldots, \pi^\Delta_{y_l}] \circ \pi^\Delta \\
&= [I_\Delta \circ \rho^\Delta, \pi^\Delta_{y_1} \circ \rho^\Delta, \ldots, \pi^\Delta_{y_l} \circ \rho^\Delta] \\
&= [1_\mathbf{1}, \rho(y_1), \ldots, \rho(y_l)] \\
&= \rho^\Gamma. \ \Box
\end{aligned}$$

3.3.17. DEFINITION. Let $[\![\ ]\!]^* : \mathcal{T}(\mathbb{C}) \times Val(\mathfrak{M}) \to |\mathfrak{M}|$ be defined as follows. Let $M \in \mathcal{T}(\mathbb{C})$, $\rho \in Val(\mathfrak{M})$. Let Δ be such that $M \in \mathcal{T}(\mathbb{C})^\Delta$. Then define

$$[\![M]\!]^*_\rho = [\![M]\!]_\Delta \circ \rho^\Delta. \ \Box$$

By 3.3.8 and (ii) of the previous lemma it follows that $[\![M]\!]_\Delta \circ \rho^\Delta$ is independent of Δ.

3.3.18. LEMMA. *$[\![\ ]\!]^*$ is a $\mathcal{T}(\mathbb{C})$-interpretation in $\mathfrak{M}_\mathcal{C}(a)$.*

PROOF. (i) $[\![x]\!]^*_\rho = [\![x]\!]_\Delta \circ \rho^\Delta = \pi^\Delta_x \circ \rho^\Delta = \rho(x)$.
(ii)

$$\begin{aligned}
[\![MN]\!]^*_\rho &= [\![MN]\!]^*_\Delta \circ \rho^\Delta \\
&= ev \circ [[\![M]\!]_\Delta, [\![N]\!]_\Delta] \circ \rho^\Delta \\
&= ev \circ [[\![M]\!] \circ \rho^\Delta, [\![N]\!]_\Delta \circ \rho^\Delta] \\
&= ev \circ [[\![M]\!]^*_\rho, [\![N]\!]^*_\rho] \\
&= [\![M]\!]^*_\rho [\![N]\!]^*_\rho.
\end{aligned}$$

(iii) Say $x \in Var_\sigma$. Then for all $p \in \mathfrak{M}_\sigma$

$$\begin{aligned}
[\![\lambda x.M]\!]^*_\rho p &= ev \circ [[\![\lambda x.M]\!]^*_\rho, p] \\
&= ev \circ [[\![\lambda x.M]\!]_\Delta \circ \rho^\Delta, p] \\
&= ev \circ [\Lambda([\![M]\!]_{\Delta,x}) \circ \rho^\Delta, p] \\
&= ev \circ [\Lambda([\![M]\!]_{\Delta,x}) \circ \rho^\Delta, 1_{\mathbf{m}_\sigma} \circ p] \\
&= ev \circ (\Lambda([\![M]\!]_{\Delta,x}) \times 1_{\mathbf{m}_\sigma}) \circ [\rho^\Delta, p] \\
&= [\![M]\!]_{\Delta,x} \circ [\rho^\Delta, p] \\
&= [\![M]\!]_{\Delta,x} \circ \rho(x := p)^{\Delta,x} \\
&= [\![M]\!]^*_{\rho(x:=p)}.
\end{aligned}$$

(iv) Let $\rho_1 =_M \rho_2$. Let $FV(M) = \{x_1, \ldots, x_k\}$ and let $\Delta = (x_1, \ldots, x_k)$. Then $\rho_1^\Delta = \rho_2^\Delta$ and so

$$[\![M]\!]^*_{\rho_1} = [\![M]\!]_\Delta \circ \rho_1^\Delta = [\![M]\!]_\Delta \circ \rho_2^\Delta = [\![M]\!]^*_{\rho_2}. \;\square$$

3.3.19. THEOREM. *$(\mathfrak{M}_\mathcal{C}(a), [\![\;]\!]^*)$ is a $\lambda^\to\eta(\mathbb{C})$-algebra.*

PROOF. Let $\lambda^\to\eta(\mathbb{C}) \vdash M = N$. Let $\rho \in Val(\mathfrak{M})$. Let Δ be such that $M, N \in \mathcal{T}(\mathbb{C})^\Delta$. Then $[\![M]\!]_\Delta = [\![N]\!]_\Delta$ by 3.3.14. So

$$[\![M]\!]^*_\rho = [\![M]\!]_\Delta \circ \rho^\Delta = [\![N]\!]_\Delta \circ \rho^\Delta = [\![N]\!]^*_\rho.$$

Therefore $\mathfrak{M}_\mathcal{C}(a), [\![\;]\!]^* \models M = N$. $\square$

Recall that $f \approx g$ denotes extensional equality of morphisms in $\mathcal{C}$.

3.3.20. PROPOSITION. *For every Δ, every $\sigma \in \mathbb{T}$ and all $M, N \in \mathcal{T}_\sigma(\mathbb{C})^\Delta$*

$$\mathfrak{M}_\mathcal{C}(a), [\![\;]\!]^* \models M = N \iff [\![M]\!]_\Delta \approx [\![N]\!]_\Delta.$$

PROOF. The map $\rho \mapsto \rho^\Delta$ is a surjection from $Val(\mathfrak{M})$ onto $|(\mathbf{m})_\Delta|$, and therefore we have

$$\begin{aligned}
\mathfrak{M}_\mathcal{C}(a), [\![\;]\!]^* \models M = N &\Leftrightarrow \forall\rho[\, [\![M]\!]^*_\rho = [\![N]\!]^*_\rho \,] \\
&\Leftrightarrow \forall\rho[\, [\![M]\!]_\Delta \circ \rho^\Delta = [\![N]\!]_\Delta \circ \rho^\Delta \,] \\
&\Leftrightarrow \forall p[\, [\![M]\!]_\Delta \circ p = [\![N]\!]_\Delta \circ p \,] \\
&\Leftrightarrow [\![M]\!]_\Delta \approx [\![N]\!]_\Delta. \;\square
\end{aligned}$$

3.3.21. THEOREM. *If $\mathcal{C}$ is λ-extensional, then $(\mathfrak{M}_{\mathcal{C}}(a), [\![\,]\!]^*)$ is a $\lambda^{\to}\eta(\mathbb{C})$-model.*

PROOF. Let $\mathcal{C}$ be λ-extensional. This means that the operator Λ is extensional. We have to show that $Th(\mathfrak{M}_{\mathcal{C}}(a), [\![\,]\!]^*)$ is a $\lambda^{\to}\eta(\mathbb{C})$-theory. For this we only need to prove that

$$\mathfrak{M}_{\mathcal{C}}(a), [\![\,]\!]^* \models M = N \Rightarrow \mathfrak{M}_{\mathcal{C}}(a), [\![\,]\!]^* \models \lambda x.M = \lambda x.N.$$

So suppose $\mathfrak{M}_{\mathcal{C}}(a), [\![\,]\!]^* \models M = N$. Let Δ be such that $(\lambda x.M), (\lambda x.N) \in \mathcal{T}(\mathbb{C})^{\Delta}$. Then $M, N \in \mathcal{T}(\mathbb{C})^{\Delta,x}$ and so $[\![M]\!]_{\Delta,x} \approx [\![N]\!]_{\Delta,x}$ by the previous proposition. By the extensionality of Λ this gives

$$[\![\lambda x.M]\!]_{\Delta} = \Lambda([\![M]\!]_{\Delta,x}) \approx \Lambda([\![N]\!]_{\Delta,x}) = [\![\lambda x.N]\!]_{\Delta}.$$

So $\mathfrak{M}_{\mathcal{C}}(a), [\![\,]\!]^* \models \lambda x.M = \lambda x.N$. So $Th(\mathfrak{M}_{\mathcal{C}}(a), [\![\,]\!]^*)$ is a theory, and so, by 3.3.14 this is a $\lambda^{\to}\eta(\mathbb{C})$-theory. Therefore $(\mathfrak{M}_{\mathcal{C}}(a), [\![\,]\!]^*)$ is a $\lambda^{\to}\eta(\mathbb{C})$-model. $\Box$

3.3.22. COROLLARY. *If $\mathcal{C}$ is extensional, then $(\mathfrak{M}_{\mathcal{C}}(a), [\![\,]\!]^*)$ is a $\lambda^{\to}\eta(\mathbb{C})$-model.* $\Box$

3.4. Complete partial orders.

Complete partial orders together with monotone functions form an extensional cartesian closed category **Mon**. This section is devoted to the category **Mon**.

3.4.1. DEFINITION. Let D be a set. Let $\sqsubseteq\subseteq D \times D$ be a binary relation on D. The relation $\sqsubseteq$ is called a partial ordering on D if for all $a, b, c \in D$

(i) $a \sqsubseteq a$ (reflexivity)
(ii) $a \sqsubseteq b \,\&\, b \sqsubseteq a \Rightarrow a = b$ (anti-symmetry)
(iii) $a \sqsubseteq b \,\&\, b \sqsubseteq c \Rightarrow a \sqsubseteq c$ (transitivity).

If $\sqsubseteq$ is a partial ordering on D, then the structure $\mathbb{D} = (D, \sqsubseteq)$ is called a partial order. $\Box$

In the following $\mathbb{D} = (D, \sqsubseteq)$ is a partial order.

3.4.2. DEFINITION. Let $X \subseteq D$.

(i) An element $a \in D$ is an upper bound of X (notation $X \sqsubseteq a$) if $b \sqsubseteq a$ for all $b \in X$.

(ii) An element $a \in D$ is a supremum of X if $X \sqsubseteq a$ and for every $b \in D$

$$X \sqsubseteq b \Rightarrow a \sqsubseteq b. \ \Box$$

Every subset X of D can have at most one supremum. If $X \subseteq D$ has a supremum then this is called the supremum of X and is denoted by $\mathbf{sup}(X)$ or by $\mathbf{sup}X$. So $\mathbf{sup}$ is a partial function from $\wp(D)$ into D.

3.4.3. DEFINITION. (i) Let $X \subseteq D$. Then X is directed if

$$\forall a, b \in X \exists c \in X[\, a \sqsubseteq c \,\&\, b \sqsubseteq c\,].$$

(ii) The partial order $\mathbb{D} = (D, \sqsubseteq)$ is complete if every directed subset of D has a supremum. A complete partial order is called a cpo. $\square$

Note that the empty set $\emptyset$ is directed. Therefore every cpo has a least element $\mathbf{sup}(\emptyset) = \bot$ (bottom).

3.4.4. EXAMPLES. Let A and B be sets.
(i) $(\wp(A), \subseteq)$ is a cpo, with $\mathbf{sup}(X) = \bigcup X$.
(ii) $\mathbb{P}(A, B)$ is the set of all partial functions from A into B. Then $(\mathbb{P}(A, B), \subseteq)$ is a cpo, with $\mathbf{sup}(X) = \bigcup X$. $\square$

3.4.5. PROPOSITION. *Let A be a set. Let $\bot$ be some object with $\bot \notin A$. Define $A_\bot = A \cup \{\bot\}$. For all $a, b \in A_\bot$ define*

$$a \sqsubseteq b \iff a = \bot \ \vee \ a = b.$$

Then $\mathbb{A}_\bot = (A_\bot, \sqsubseteq)$ is a cpo, called the flat cpo over A.

PROOF. Let $X \subseteq A_\bot$ be directed. If there exists $a \in A$ with $a \in X$, then by the directedness of X it follows that $X = \{\bot, a\}$ or $X = \{a\}$, so $\mathbf{sup}(X) = a$. On the other hand, if $X \cap A = \emptyset$, then $\mathbf{sup}(X) = \bot$. $\square$

In the next section the flat cpo's $\mathbb{A}_\bot$ will be used to construct extensional $\lambda\mathsf{Y}\eta(\mathcal{A})$-models.

In the following $\mathbb{D}$, $\mathbb{D}_1, \mathbb{D}_2, \mathbb{D}_3, \ldots$ are cpo's, with $\mathbb{D} = (D, \sqsubseteq)$ and $\mathbb{D}_n = (D_n, \sqsubseteq_n)$ for $n = 1, 2, 3, \ldots$

3.4.6. DEFINITION. (i) A function $f : D_1 \to D_2$ is monotone if for all $a, b \in D_1$

$$a \sqsubseteq_1 b \Rightarrow f(a) \sqsubseteq_2 f(b).$$

If $f : D_1 \to D_2$ is monotone, then we write $f : \mathbb{D}_1 \to \mathbb{D}_2$.
(ii) $\mathbf{Mon}[\mathbb{D}_1, \mathbb{D}_2]$ is the set of all monotone functions from D_1 into D_2.
(iii) For all $f, g \in \mathbf{Mon}[\mathbb{D}_1, \mathbb{D}_2]$ define

$$f \sqsubseteq g \iff \forall a \in D_1[\, f(a) \sqsubseteq_2 g(a)\,].$$

(iv) $\mathbf{Mon}(\mathbb{D}_1, \mathbb{D}_2) = (\mathbf{Mon}[\mathbb{D}_1, \mathbb{D}_2], \sqsubseteq)$. $\square$

We will show that the partial order $\mathbf{Mon}(\mathbb{D}_1, \mathbb{D}_2)$ is again complete.

3.4.7. DEFINITION. For $X \subseteq \mathbf{Mon}[\mathbb{D}_1, \mathbb{D}_2]$ and $a \in D_1$ define

$$X(a) = \{ f(a) | \, f \in X \}. \; \square$$

3.4.8. LEMMA. *If $X \subseteq \mathbf{Mon}[\mathbb{D}_1, \mathbb{D}_2]$ is directed, then $X(a) \subseteq D_2$ is directed for every $a \in D_1$.*

PROOF. Let $X \subseteq \mathbf{Mon}[\mathbb{D}_1, \mathbb{D}_2]$ be directed, and let $a \in D_1$. Let $p, q \in X(a)$. Then there exist $f, g \in X$ with $p = f(a)$ and $q = g(a)$. Because X is directed, there exists $h \in X$ with $f \sqsubseteq h$ and $g \sqsubseteq h$. Define $r = h(a)$. Then $r \in X(a)$ and $p \sqsubseteq r$ and $q \sqsubseteq r$. So $X(a)$ is directed. $\square$

3.4.9. LEMMA. *Let $X \subseteq \mathbf{Mon}[\mathbb{D}_1, \mathbb{D}_2]$ be directed. Define $f : D_1 \to D_2$ by*

$$f(a) = \mathbf{sup} X(a).$$

Then $f \in \mathbf{Mon}[\mathbb{D}_1, \mathbb{D}_2]$ and $f = \mathbf{sup}(X)$.

PROOF. (i) First we prove monotonicity. Let $a \sqsubseteq_1 b$. Then for every $g \in X$ we have $g(a) \sqsubseteq_2 g(b) \sqsubseteq_2 f(b)$ because g is monotone and $g(b) \in X(b) \sqsubseteq_2 f(b)$. So $X(a) \sqsubseteq_2 f(b)$, so $f(a) = \mathbf{sup} X(a) \sqsubseteq_2 f(b)$. So f is monotone and $f \in \mathbf{Mon}[\mathbb{D}_1, \mathbb{D}_2]$.

(ii) Now we prove $f = \mathbf{sup}(X)$. First we show $X \sqsubseteq f$. So let $g \in X$. Let $a \in D_1$. Then $g(a) \in X(a)$ and $X(a) \sqsubseteq_2 f(a)$. So $g(a) \sqsubseteq_2 f(a)$. So $g \sqsubseteq f$. Therefore $X \sqsubseteq f$.

Now suppose $X \sqsubseteq g$. Let $a \in D_1$. Then $X(a) \sqsubseteq_2 g(a)$, so $f(a) = \mathbf{sup} X(a) \sqsubseteq_2 g(a)$. So $f \sqsubseteq g$.

It follows that $f = \mathbf{sup}(X)$. $\square$

3.4.10. PROPOSITION. (i) *$\mathbf{Mon}(\mathbb{D}_1, \mathbb{D}_2)$ is a cpo.*

(ii) *For every directed $X \subseteq \mathbf{Mon}[\mathbb{D}_1, \mathbb{D}_2]$ we have*

$$(\mathbf{sup} X)(a) = \mathbf{sup} X(a)$$

for all $a \in D_1$.

PROOF. By the previous lemma. $\square$

Next it will be shown that every monotone $f : D \to D$ has a least fixed point. This is the Knaster-Tarski Fixed Point Theorem.

Let $f : D \to D$ be monotone. By transfinite induction we define, for every $\zeta \in \mathbf{On}$, an element $f^\zeta \in D$ by

$$f^\zeta = f(\mathbf{sup}\{ f^\xi | \, \xi < \zeta \}).$$

By transfinite induction it can be shown that every f^ζ is well-defined, by proving that every set
$$X_\zeta = \{f^\xi \mid \xi < \zeta\}$$
is a chain and therefore directed, so that $\mathbf{sup}(X_\zeta)$ exists.

For every $\zeta \in \mathbf{On}$ we set
$$f^{(\zeta)} = \mathbf{sup}\{f^\xi \mid \xi < \zeta\}$$
so that we have $f^\zeta = f(f^{(\zeta)})$.

3.4.11. Lemma. *For all* $\zeta, \xi \in \mathbf{On}$
$$\zeta \leq \xi \Rightarrow f^\zeta \sqsubseteq f^\xi.$$

Proof. Let $\zeta \leq \xi$. Then $X_\zeta \subseteq X_\xi$, so $f^{(\zeta)} = \mathbf{sup}(X_\zeta) \sqsubseteq \mathbf{sup}(X_\xi) = f^{(\xi)}$. Because f is monotone this gives $f^\zeta \sqsubseteq f^\xi$. □

3.4.12. Lemma. *Let* $\zeta \in \mathbf{On}$.
(i) $f^{(\zeta+1)} = f^\zeta$.
(ii) $f^{\zeta+1} = f(f^\zeta)$.

Proof. (i) $f^\zeta \in X_{\zeta+1} \sqsubseteq f^\zeta$, so $f^\zeta = \mathbf{sup}(X_{\zeta+1}) = f^{(\zeta+1)}$.
(ii) $f^{\zeta+1} = f(f^{(\zeta+1)}) = f(f^\zeta)$ by (i). □

3.4.13. Proposition. *There exists* $\zeta \in \mathbf{On}$ *such that* $\mathbf{card}(\zeta) \leq \mathbf{card}(D)$ *and* $f(f^\zeta) = f^\zeta$.

Proof. Let $\kappa = \mathbf{card}(D)^+$ (the first cardinal number after $\mathbf{card}(D)$). Suppose, towards a contradiction $\forall \zeta < \kappa [\, f^{\zeta+1} \neq f^\zeta\,]$. Let $\zeta < \xi < \kappa$. We want to show $f^\zeta \neq f^\xi$. So suppose $f^\zeta = f^\xi$. Then
$$f^\zeta \sqsubseteq f^{\zeta+1} \sqsubseteq f^\xi = f^\zeta$$
so $f^\zeta = f^{\zeta+1}$. Contradiction. So $f^\zeta \neq f^\xi$. Therefore we have that for all $\zeta, \xi < \kappa$
$$\zeta \neq \xi \Rightarrow f^\zeta \neq f^\xi.$$
So $\{f^\zeta \mid \zeta < \kappa\}$ is a subset of D with cardinality κ. Contradiction. So there exists $\zeta < \kappa$ with $f^{\zeta+1} = f^\zeta$. Then $\mathbf{card}(\zeta) \leq \mathbf{card}(D)$ and $f(f^\zeta) = f^\zeta$. □

3.4.14. Definition. (i) The closure ordinal of f is defined by
$$|f| = \mu\zeta[\, f(f^\zeta) = f^\zeta\,].$$

(ii) $f^\infty = f^{|f|}$. □

It follows that $f(f^\infty) = f^\infty$ and $\mathbf{card}(|f|) \leq \mathbf{card}(D)$. In the following it will be shown that f^∞ is the least fixed point of f, and it will be shown that the mapping $f \mapsto f^\infty$ from $\mathbf{Mon}[\mathbb{D}, \mathbb{D}]$ into D is monotone.

3.4.15. LEMMA. $\forall a \in D[\, f(a) \sqsubseteq a \Rightarrow f^\infty \sqsubseteq a\,]$.

PROOF. Let $a \in D$ with $f(a) \sqsubseteq a$. By transfinite induction we show $f^\zeta \sqsubseteq a$ for all $\zeta \in \mathbf{On}$. Let $\zeta \in \mathbf{On}$ and suppose $\forall \xi < \zeta[\, f^\xi \sqsubseteq a\,]$. Then $\{f^\xi | \,\xi < \zeta\,\} \sqsubseteq a$, so $f^{(\zeta)} \sqsubseteq a$ and so $f^\zeta = f(f^{(\zeta)}) \sqsubseteq f(a) \sqsubseteq a$.

Therefore $\forall \zeta \in \mathbf{On}[\, f^\zeta \sqsubseteq a\,]$, and so $f^\infty = f^{|f|} \sqsubseteq a$. □

3.4.16. THEOREM (Knaster-Tarski Fixed Point Theorem).
For every monotone $f : D \to D$ we have that f^∞ is the least fixed point of f.

PROOF. Let $f : D \to D$ be monotone. First note that f^∞ is a fixed point of f. Now let $a \in D$ be any fixed point of f. Then $f(a) \sqsubseteq a$ and so $f^\infty \sqsubseteq a$, by lemma 3.4.15. □

3.4.17. THEOREM. *Define* $\mathbf{fix}_{\mathbb{D}} : \mathbf{Mon}[\mathbb{D}, \mathbb{D}] \to D$ *by*

$$\mathbf{fix}_{\mathbb{D}}(f) = f^\infty.$$

Then $\mathbf{fix}_{\mathbb{D}} \in \mathbf{Mon}[\mathbf{Mon}(\mathbb{D}, \mathbb{D}), \mathbb{D}]$.

PROOF. Let $f, g : D \to D$ be monotone with $f \sqsubseteq g$. Then $f(g^\infty) \sqsubseteq g(g^\infty) = g^\infty$, so by lemma 3.4.15 we have $f^\infty \sqsubseteq g^\infty$. So $\mathbf{fix}_{\mathbb{D}}$ is monotone. □

Sets that are inductively defined by monotone operators are special cases of least fixed points of monotone functions on cpo's.

Let A be a non-empty set. Recall that $(\wp(A), \subseteq)$ is a cpo with $\mathbf{sup}(X) = \bigcup X$ for every $X \subseteq \wp(A)$. Let $\Phi : \wp(A) \to \wp(A)$ be a monotone operator on A. Then for all $\zeta \in \mathbf{On}$

$$\Phi^{(\zeta)} = \bigcup_{\xi < \zeta} \Phi^\xi$$

so for all $\zeta \in \mathbf{On}$

$$\Phi^\zeta = \Phi(\bigcup_{\xi < \zeta} \Phi^\xi).$$

The least fixed point

$$\Phi^\infty = \bigcup_{\zeta \in \mathbf{On}} \Phi^\zeta$$

of Φ is called the subset of A which is inductively defined by the operator Φ.

3.4.18. DEFINITION. $\mathbf{Mon}$ is the category with cpo's as objects and monotone functions as morphisms. □

Because the composition of monotone functions is again monotone, and because identity functions on cpo's are monotone, **Mon** is indeed a category. We are going to show that **Mon** is an extensional cartesian closed category.

3.4.19. Lemma. *Let* $\mathbf{1}$ *be the singleton cpo with domain* $1 = \{0\}$*. Then* $\mathbf{1}$ *is a terminal object of* **Mon**.

Proof. Let $I_{\mathbb{D}} : D \to \mathbf{1}$ be defined by $I_{\mathbb{D}}(a) = 0$. Then $I_{\mathbb{D}}$ is monotone, and for every $f : \mathbb{D} \to \mathbf{1}$ we have $f = I_{\mathbb{D}}$. So $\mathbf{1}$ is terminal in **Mon**. □

The points of a cpo $\mathbb{D} = (D, \sqsubseteq)$ correspond exactly with the elements of D. For every $a \in D$ define $p_a : 1 \to D$ by $p_a(0) = a$. Then $p_a : \mathbf{1} \to \mathbb{D}$, so p_a is a point of $\mathbb{D}$. Also define $P_{\mathbb{D}} : D \to |\mathbb{D}|$ by $P_{\mathbb{D}}(a) = p_a$.

3.4.20. Lemma. (i) $\forall a, b \in D[\, p_a = p_b \Rightarrow a = b\,]$.
(ii) $\forall p \in |\mathbb{D}| \exists a \in D[\, p = p_a\,]$.
(iii) $P_{\mathbb{D}} : D \rightarrowtail\!\!\!\!\rightarrow |\mathbb{D}|$.

Proof. (i) Let $a, b \in D$ with $p_a = p_b$. Then $a = p_a(0) = p_b(0) = b$.
(ii) Let $p \in |\mathbb{D}|$. Define $a = p(0)$. Then $p_a(0) = a = p(0)$, so $p_a = p$.
(iii) By (i) and (ii). □

3.4.21. Lemma. *Let* $f : \mathbb{D}_1 \to \mathbb{D}_2$*. Then for all* $a \in D_1$

$$|f|(p_a) = p_{f(a)}.$$

Proof. Let $a \in D_1$. Then $|f|(p_a) = f \circ p_a : \mathbf{1} \to \mathbb{D}_2$ and $(f \circ p_a)(0) = f(p_a(0)) = f(a) = p_{f(a)}(0)$, so $f \circ p_a = p_{f(a)}$ and so $|f|(p_a) = p_{f(a)}$. □

3.4.22. Corollary. **Mon** *is extensional.*

Proof. Let $f, g : \mathbb{D}_1 \to \mathbb{D}_2$ be such that $f \approx g$. Then $|f| = |g|$. Let $a \in D_1$. Then $p_{f(a)} = |f|(p_a) = |g|(p_a) = p_{g(a)}$ so $f(a) = g(a)$. Therefore we have $f = g$. So **Mon** is extensional. □

In the following it will be shown that **Mon** has products and exponentiation.

First we show that **Mon** has products. We make $D_1 \times D_2$ into a partial order by defining

$$(a_1, a_2) \sqsubseteq (b_1, b_2) \Longleftrightarrow a_1 \sqsubseteq_1 b_1 \;\&\; a_2 \sqsubseteq_2 b_2.$$

For $X \subseteq D_1 \times D_2$ define

$$(X)_1 = \{a_1 \in D_1 |\, \exists a_2 \in D_2[\, (a_1, a_2) \in X\,]\}$$

$$(X)_2 = \{a_2 \in D_2 |\, \exists a_1 \in D_1[\, (a_1, a_2) \in X\,]\}.$$

If $X \subseteq D_1 \times D_2$ is directed, then $(X)_1$ and $(X)_2$ are also directed, and

$$\mathbf{sup}(X) = (\mathbf{sup}(X)_1, \mathbf{sup}(X)_2).$$

Therefore $(D_1 \times D_2, \sqsubseteq)$ is a cpo. This cpo is denoted by $\mathbb{D}_1 \times \mathbb{D}_2$. The projections $\pi_i : D_1 \times D_2 \to D_i$ $(i = 1, 2)$ are defined by

$$\pi_i(a_1, a_2) = a_i.$$

Then $\pi_i : \mathbb{D}_1 \times \mathbb{D}_2 \to \mathbb{D}_i$ for $i = 1, 2$. Pairs of morphisms are defined as follows. Let $f : \mathbb{D} \to \mathbb{D}_1$ and $g : \mathbb{D} \to \mathbb{D}_2$. Define $[f, g] : D \to D_1 \times D_2$ by

$$[f, g](a) = (f(a), g(a)).$$

Then $[f, g] : \mathbb{D} \to \mathbb{D}_1 \times \mathbb{D}_2$ and for every $h : \mathbb{D} \to \mathbb{D}_1 \times \mathbb{D}_2$ we have

$$h = [f, g] \Longleftrightarrow \pi_1 \circ h = f \ \& \ \pi_2 \circ h = g.$$

This shows that **Mon** has products.

Let $f : \mathbb{D}_1 \to \mathbb{D}_3$ and $g : \mathbb{D}_2 \to \mathbb{D}_4$. Then for

$$f \times g = [f \circ \pi_1, g \circ \pi_2] : \mathbb{D}_1 \times \mathbb{D}_2 \to \mathbb{D}_3 \times \mathbb{D}_4$$

we have that for all $(a_1, a_2) \in D_1 \times D_2$

$$\begin{aligned} (f \times g)(a_1, a_2) &= [f \circ \pi_1, g \circ \pi_2](a_1, a_2) \\ &= ((f \circ \pi_1)(a_1, a_2), (g \circ \pi_2)(a_1, a_2)) \\ &= (f(\pi_1(a_1, a_2)), g(\pi_2(a_1, a_2))) \\ &= (f(a_1), g(a_2)) \end{aligned}$$

so $(f \times g)(a_1, a_2) = (f(a_1), g(a_2))$.

Next we show that **Mon** has exponentiation. The exponent of $\mathbb{D}_1$ and $\mathbb{D}_2$ is the cpo $\mathbf{Mon}(\mathbb{D}_1, \mathbb{D}_2)$. The evaluation morphism

$$ev : \mathbf{Mon}(\mathbb{D}_1, \mathbb{D}_2) \times \mathbb{D}_1 \to \mathbb{D}_2$$

is defined by $ev(f, a) = f(a)$.

Let $f : \mathbb{D} \times \mathbb{D}_1 \to \mathbb{D}_2$. Then $\Lambda(f) : D \to \mathbf{Mon}(\mathbb{D}_1, \mathbb{D}_2)$ is defined as follows. For every $a \in D$ let $\Lambda(f)(a)$ be the monotone map $a_1 \mapsto f(a, a_1)$ from D_1 into D_2. Then $\Lambda(f) : \mathbb{D} \to \mathbf{Mon}(\mathbb{D}_1, \mathbb{D}_2)$ and $ev \circ (\Lambda(f) \times 1_{\mathbb{D}_1}) = f$ and for every $g : \mathbb{D} \to \mathbf{Mon}(\mathbb{D}_1, \mathbb{D}_2)$ we have

$$ev \circ (g \times 1_{\mathbb{D}_1}) = f \Rightarrow f = g.$$

This shows that **Mon** has exponentiation.

3.4.23. Theorem. **Mon** *is an extensional cartesian closed category.* $\square$

3.5. Monotone $\lambda\mathsf{Y}\eta(\mathcal{A})$-models.

Let $\mathcal{A}$ be a structure with domain A. In this section we will look at a condition on cpo's $\mathbb{D}$ under which the type structure $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ can be made into a $\lambda\mathsf{Y}\eta(\mathcal{A})$-model. In particular the flat cpo $\mathbb{A}_\perp$ over A will satisfy this condition, so that the type structure $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{A}_\perp)$ is a pre-$\lambda\mathsf{Y}\eta(\mathcal{A})$-model.

Let $\mathbb{D} = (D, \sqsubseteq)$ be a cpo. Let us review how the type structure $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ is obtained from $\mathbb{D}$. First define, for every $\sigma \in \mathbb{T}$, a cpo $\mathbb{D}_\sigma$ by induction on the structure of σ as follows.

- $\mathbb{D}_0 = \mathbb{D}$
- $\mathbb{D}_{\sigma\to\tau} = \mathbf{Mon}(\mathbb{D}_\sigma, \mathbb{D}_\tau)$.

For all $\sigma, \tau \in \mathbb{T}$ let $App_{\sigma,\tau} : \mathbb{D}_{\sigma\to\tau} \times \mathbb{D}_\sigma \to \mathbb{D}_\tau$ be the evaluation $\mathbf{Mon}(\mathbb{D}_\sigma, \mathbb{D}_\tau) \times \mathbb{D}_\sigma \to \mathbb{D}_\tau$. Then

$$\mathbf{m}_{\mathbf{Mon}}(\mathbb{D}) = ((\mathbb{D}_\sigma)_{\sigma\in\mathbb{T}}, (App_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}})$$

is the type structure in **Mon** which is generated from $\mathbb{D}$. The type structure $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ is now obtained from $\mathbf{m}_{\mathbf{Mon}}(\mathbb{D})$ by applying the functor $^* : \mathbf{TS}(\mathbf{Mon}) \to \mathbf{TS}$:

$$\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D}) = \mathbf{m}_{\mathbf{Mon}}(\mathbb{D})^* = ((\mathbb{D}^*_\sigma)_{\sigma\in\mathbb{T}}, (App^*_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}})$$

where $\mathbb{D}^*_\sigma = |\mathbb{D}_\sigma|$ is the set of points of $\mathbb{D}_\sigma$. We see that objects of $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ are points of the cpo's $\mathbb{D}_\sigma$. This is a little awkward. It is more convenient to work with the isomorphic type structure $\mathfrak{M}$ defined as follows. Say $\mathbb{D}_\sigma = (D_\sigma, \sqsubseteq_\sigma)$ for every $\sigma \in \mathbb{T}$. Then for all $\sigma, \tau \in \mathbb{T}$ we have that $D_{\sigma\to\tau}$ is the set of all monotone functions from $\mathbb{D}_\sigma$ into $\mathbb{D}_\tau$, so $D_{\sigma\to\tau} = \mathbf{Mon}[\mathbb{D}_\sigma, \mathbb{D}_\tau]$, and for all $f, g \in D_{\sigma\to\tau}$

$$f \sqsubseteq_{\sigma\to\tau} g \iff \forall a \in D_\sigma [\, f(a) \sqsubseteq_\tau g(a)\,].$$

Note that for the application function $App_{\sigma,\tau}$ of $\mathbf{m}_{\mathbf{Mon}}(\mathbb{D})$ we have that $App_{\sigma,\tau} : D_{\sigma\to\tau} \times D_\sigma \to D_\tau$ is a monotone function such that for all $f \in D_{\sigma\to\tau}$ and $a \in D_\sigma$

$$App_{\sigma,\tau}(f, a) = f(a)$$

because evaluation in **Mon** is function application. Now define

$$\mathfrak{M} = ((D_\sigma)_{\sigma\in\mathbb{T}}, (App_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}).$$

First we will show that $\mathfrak{M}$ is isomorphic with $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$. Recall that $\mathbf{1}$ is the singleton cpo with domain $1 = \{0\}$. For all $\sigma \in \mathbb{T}$ and $a \in D_\sigma$ we have the point $p_a : \mathbf{1} \to \mathbb{D}_\sigma$ of $\mathbb{D}_\sigma$ defined by $p_a(0) = a$.

3.5.1. LEMMA. *For all $\sigma, \tau \in \mathbb{T}$, $a \in D_{\sigma\to\tau}$ and $b \in D_\sigma$*

$$p_a p_b = p_{ab}.$$

PROOF. Let $\sigma, \tau \in \mathbb{T}$, $a \in D_{\sigma\to\tau}$, $b \in D_\sigma$. Then $a : D_\sigma \to D_\tau$ and $ab = a(b)$. Note that $p_a p_b$ and p_{ab} are functions from $1 = \{0\}$ into D_τ. We have

$$\begin{aligned}(p_a p_b)(0) &= (ev \circ [p_a, p_b])(0)\\ &= ev([p_a, p_b](0))\\ &= ev(p_a(0), p_b(0))\\ &= ev(a, b)\\ &= a(b)\\ &= ab\\ &= p_{ab}(0)\end{aligned}$$

so $p_a p_b = p_{ab}$. $\Box$

3.5.2. COROLLARY. $\mathfrak{M} \cong \mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$.

PROOF. For every $\sigma \in \mathbb{T}$ define $P_\sigma : D_\sigma \to \mathbb{D}^*_\sigma$ by $P_\sigma(a) = p_a$. Then define $P = (P_\sigma)_{\sigma\in\mathbb{T}}$. By the previous lemma P is a homomorphism from $\mathfrak{M}$ into $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$. But every P_σ is a bijection. So P is an isomorphism. $\Box$

In the following we will identify $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ with $\mathfrak{M}$, so

$$\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D}) = ((D_\sigma)_{\sigma\in\mathbb{T}}, (App_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}).$$

The type structure $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ is called the monotone type structure over $\mathbb{D}$. Because **Mon** is an extensional ccc, $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ is an extensional type structure. By the results of section 3.3 $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ carries $\mathcal{T}(\mathbb{C})$-interpretations. Therefore $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ is an extensional λ-structure.

3.5.3. LEMMA. *For every $\sigma \in \mathbb{T}$ and all $a, b \in D_\sigma$*

$$a \sqsubseteq_\sigma b \Longleftrightarrow \forall \vec{c} \in (\mathfrak{M})_\sigma[\, a\vec{c} \sqsubseteq b\vec{c}\,].$$

PROOF. By induction on the structure of σ.

(i) $\sigma = 0$: simple.

(ii) Let $a, b \in D_{\sigma\to\tau}$.

(1) Suppose $a \sqsubseteq_{\sigma\to\tau} b$. Let $(c, \vec{d}) \in (\mathfrak{M})_{\sigma\to\tau} = \mathfrak{M}_\sigma \times (\mathfrak{M})_\tau$. Then $ac = a(c) \sqsubseteq_\tau b(c) = bc$. So by induction $ac\vec{d} \sqsubseteq bc\vec{d}$.

(2) Suppose $\forall (c, \vec{d}) \in (\mathfrak{M})_{\sigma\to\tau}[\, ac\vec{d} \sqsubseteq bc\vec{d}\,]$. Let $c \in D_\sigma = \mathfrak{M}_\sigma$. Then $\forall \vec{d} \in (\mathfrak{M})_\tau[\, ac\vec{d} \sqsubseteq bc\vec{d}\,]$. So by induction $ac \sqsubseteq_\tau bc$. So $\forall c \in D_\sigma[\, a(c) \sqsubseteq_\tau b(c)\,]$ and therefore $a \sqsubseteq_{\sigma\to\tau} b$. $\Box$

Instead of $a \sqsubseteq_\sigma b$ we will usually write $a \sqsubseteq b$. For $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0$, $\vec{a} = (a_1, \ldots, a_k) \in (\mathfrak{M})_\sigma$ and $\vec{b} = (b_1, \ldots, b_k) \in (\mathfrak{M})_\sigma$

$$\vec{a} \sqsubseteq \vec{b} \Longleftrightarrow a_i \sqsubseteq b_i \ \ (i = 1, \ldots, k).$$

3.5.4. LEMMA. *For all $\sigma \in \mathbb{T}$, $a \in \mathfrak{M}_\sigma$, $\vec{b}, \vec{c} \in (\mathfrak{M})_\sigma$*

$$\vec{b} \sqsubseteq \vec{c} \Rightarrow a\vec{b} \sqsubseteq a\vec{c}.$$

PROOF. By induction on the structure of σ.

(i) $\sigma = 0$: simple.

(ii) Let $a \in \mathfrak{M}_{\sigma\to\tau}$, $(b_0, \vec{b}), (c_0, \vec{c}) \in (\mathfrak{M})_{\sigma\to\tau}$ with $(b_0, \vec{b}) \sqsubseteq (c_0, \vec{c})$. Then $b_0 \sqsubseteq c_0$ and $\vec{b} \sqsubseteq \vec{c}$ and $a : D_\sigma \to D_\tau$ is a monotone function, so $ab_0 \sqsubseteq ac_0$. So $ab_0\vec{b} \sqsubseteq ac_0\vec{b}$. By induction $ac_0\vec{b} \sqsubseteq ac_0\vec{c}$. And so $ab_0\vec{b} \sqsubseteq ac_0\vec{c}$. $\Box$

Next we will show that the $\mathfrak{M}$-representable functionals are precisely the monotone functionals on $\mathfrak{M}$.

3.5.5. DEFINITION. Let $\sigma \in \mathbb{T}$ and $\varphi : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$. Then φ is monotone if for all $\vec{a}, \vec{b} \in (\mathfrak{M})_\sigma$

$$\vec{a} \sqsubseteq \vec{b} \Rightarrow \varphi(\vec{a}) \sqsubseteq \varphi(\vec{b}). \ \Box$$

3.5.6. LEMMA. *For all $\sigma \in \mathbb{T}$ and $\varphi : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ we have that φ is $\mathfrak{M}$-representable if and only if φ is monotone.*

PROOF. (i) Let $\sigma \in \mathbb{T}$ and let $\varphi : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ be $\mathfrak{M}$-representable. Let $f \in \mathfrak{M}_\sigma$ $\mathfrak{M}$-represent φ. Then for all $\vec{a}, \vec{b} \in (\mathfrak{M})_\sigma$ with $\vec{a} \sqsubseteq \vec{b}$ we have $\varphi(\vec{a}) = f\vec{a} \sqsubseteq f\vec{b} = \varphi(\vec{b})$. So φ is monotone.

(ii) By induction on the structure of σ it will now be shown that for all $\sigma \in \mathbb{T}$ and all monotone $\varphi : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ we have that φ is $\mathfrak{M}$-representable.

(1) $\sigma = 0$: simple.

(2) Let $\varphi : (\mathfrak{M})_{\sigma\to\tau} \to \mathfrak{M}_0$ be monotone. For every $a \in D_\sigma$ the functional $\vec{b} \mapsto \varphi(a, \vec{b})$ from $(\mathfrak{M})_\tau$ into $\mathfrak{M}_0$ is monotone and therefore $\mathfrak{M}$-representable by induction, so by the extensionality of $\mathfrak{M}$ there exists a unique $f(a) \in D_\tau$ which $\mathfrak{M}$-represents $\vec{b} \mapsto \varphi(a, \vec{b})$. This gives a function $f : D_\sigma \to D_\tau$. We show that f is monotone as follows: Let $a_1 \sqsubseteq a_2$; then for all $\vec{b} \in (\mathfrak{M})_\tau$

$$f(a_1)\vec{b} = \varphi(a_1, \vec{b}) \sqsubseteq \varphi(a_2, \vec{b}) = f(a_2)\vec{b}$$

so $f(a_1) \sqsubseteq f(a_2)$. Therefore $f \in D_{\sigma\to\tau} = \mathfrak{M}_{\sigma\to\tau}$. For all $(a, \vec{b}) \in (\mathfrak{M})_{\sigma\to\tau}$ we have

$$\varphi(a, \vec{b}) = f(a)\vec{b} = fa\vec{b}$$

so f $\mathfrak{M}$-represents φ. $\Box$

Next it will be shown that $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ is a pre-$\lambda\mathsf{Y}\eta$-model. For every $\sigma \in \mathbb{T}$ define $\mathbb{Y}_\sigma = \mathbf{fix}_{\mathbb{D}_\sigma}$. Then $\mathbb{Y}_\sigma \in D_{(\sigma\to\sigma)\to\sigma}$ and for all $f \in D_{\sigma\to\sigma}$ we have $\mathbb{Y}_\sigma f = \mathbb{Y}_\sigma(f) = f^\infty$, so $\mathbb{Y}_\sigma f$ is the least fixed point of f.

Let $\mathbb{C}$ be such that $\mathcal{T}(\mathbb{C}) = \mathcal{T}(\lambda\mathsf{Y})$. Define $\mu : |\mathbb{C}| \to |\mathfrak{M}|$ by $\mu(\mathbf{Y}_\sigma) = \mathbb{Y}_\sigma$. Then μ is type-preserving and so there exists a $\lambda\mathsf{Y}$-interpretation $[\![\ ]\!]$ in $\mathfrak{M}$ such that $[\![\mathbf{Y}_\sigma]\!] = \mu(\mathbf{Y}_\sigma) = \mathbb{Y}_\sigma$ for all $\sigma \in \mathbb{T}$.

3.5.7. LEMMA. *$(\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D}), [\![\ ]\!])$ is a $\lambda\mathsf{Y}\eta$-model.*

PROOF. We verify the satisfaction of the γ-axioms. Let $\sigma \in \mathbb{T}$ and $x \in Var_{\sigma\to\sigma}$. Then $(\mathbf{Y}_\sigma = \lambda x.x(\mathbf{Y}_\sigma x))$ is a γ-axiom. For all $f \in D_{\sigma\to\sigma}$ we have

$$\begin{aligned}
[\![\mathbf{Y}_\sigma]\!] f &= \mathbb{Y}_\sigma f \\
&= f(\mathbf{Y}_\sigma f) \\
&= [\![x(\mathbf{Y}_\sigma x)]\!]_{\rho(x:=f)} \\
&= [\![\lambda x.x(\mathbf{Y}_\sigma x)]\!] f.
\end{aligned}$$

So $[\![\mathbf{Y}_\sigma]\!] \approx [\![\lambda x.x(\mathbf{Y}_\sigma x)]\!]$, but $\mathfrak{M}$ is extensional, so $[\![\mathbf{Y}_\sigma]\!] = [\![\lambda x.x(\mathbf{Y}_\sigma x)]\!]$. So every γ-axiom is satisfied in $(\mathfrak{M}, [\![\ ]\!])$. Therefore, by the extensionality of $\mathfrak{M}$, it follows that $(\mathfrak{M}, [\![\ ]\!])$ is a $\lambda\mathsf{Y}\eta$-model. □

Let $\mathcal{A} = (A, R_1, \ldots, R_r, f_1, \ldots, f_s)$ be a structure with similarity type $\$ = ((p_1, \ldots, p_r), (q_1, \ldots, q_s))$. This gives us the systems $\lambda\mathsf{Y}(\mathcal{A})$ and $\lambda\mathsf{Y}\eta(\mathcal{A})$. We are going to look at a condition on $\mathbb{D}$ under which $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ is a pre-$\lambda\mathsf{Y}\eta(\mathcal{A})$-model.

3.5.8. DEFINITION. Let $^* : A \to D$.

(i) If $a \in A$, then a^* is called the representative of a.

(ii) Let $m \in \omega$, $f : A^m \to A$, $\varphi : D^m \to D$. Then φ represents f w.r.t. * if for all $a_1, \ldots, a_m \in A$

$$\varphi(a_1^*, \ldots, a_m^*) = f(a_1, \ldots, a_m)^*.$$

(iii) A function $f : A^m \to A$ is $\mathbb{D}$-representable w.r.t. * if there exists a monotone function $\varphi : D^m \to D$ such that φ represents f w.r.t. *.

(iv) Let $m \in \omega$, $R \subseteq A^m$, $\varphi : D^{m+2} \to D$. Then φ represents R w.r.t. * if for all $a_1, \ldots, a_m \in A$, $b, c \in D$

(1) $\varphi(a_1^*, \ldots, a_m^*, b, c) = b$ if $R(a_1, \ldots, a_m)$

(2) $\varphi(a_1^*, \ldots, a_m^*, b, c) = c$ otherwise.

(v) A relation $R \subseteq A^m$ is $\mathbb{D}$-representable w.r.t. * if there exists a monotone function $\varphi : D^{m+2} \to D$ such that φ represents R w.r.t. *.

(vi) The structure $\mathcal{A}$ is called $\mathbb{D}$-representable w.r.t. * if every relation R_i is $\mathbb{D}$-representable w.r.t. * and every function f_j is $\mathbb{D}$-representable w.r.t. *. □

3.5.9. DEFINITION. The structure $\mathcal{A}$ is $\mathbb{D}$-representable if there exists $^* : A \to D$ such that $\mathcal{A}$ is $\mathbb{D}$-representable w.r.t. *. $\square$

In the following it will be shown that if $\mathcal{A}$ is $\mathbb{D}$-representable, then the type structure $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ is a pre-$\lambda\mathsf{Y}\eta(\mathcal{A})$-model.

So suppose $\mathcal{A}$ is $\mathbb{D}$-representable. Let $^* : A \to D$ be such that $\mathcal{A}$ is $\mathbb{D}$-representable w.r.t. *. For every $i \in \{1, \ldots, r\}$ let $\varphi_i : D^{p_i+2} \to D$ be a monotone function that represents R_i w.r.t. *, and for every $j \in \{1, \ldots, s\}$ let $\psi_j : D^{q_j} \to D$ be a monotone function that represents f_j. Then the φ_i and the ψ_j are $\mathfrak{M}$-representable functionals on $\mathfrak{M}$. Let $R_i^* \in \mathfrak{M}_{t(p_i+2)}$ $\mathfrak{M}$-represent φ_i $(i = 1, \ldots, r)$ and let $f_j^* \in \mathfrak{M}_{t(q_j)}$ $\mathfrak{M}$-represent ψ_j $(j = 1, \ldots, s)$.

Let $\mathbb{C}$ be such that $\mathcal{T}(\mathbb{C}) = \mathcal{T}(\lambda\mathsf{Y}\eta(\mathcal{A}))$, and define $\mu : |\mathbb{C}| \to |\mathfrak{M}|$ by

- $\mu(\mathbf{Y}_\sigma) = \mathbb{Y}_\sigma$ for all $\sigma \in \mathbb{T}$
- $\mu(\mathbf{c}_a) = a^*$ for all $a \in A$
- $\mu(\mathbf{R}_i) = R_i^*$ for every $i \in \{1, \ldots, r\}$
- $\mu(\mathbf{F}_j) = f_j^*$ for every $j \in \{1, \ldots, s\}$.

Now let $[\![\ ]\!]$ be the $\lambda\mathsf{Y}\eta(\mathcal{A})$-interpretation in $\mathfrak{M}$ such that $[\![\mathbf{C}]\!] = \mu(\mathbf{C})$ for all $\mathbf{C} \in |\mathbb{C}|$.

3.5.10. LEMMA. *All α-axioms and all γ-axioms of $\lambda\mathsf{Y}\eta(\mathcal{A})$ are satisfied in $(\mathfrak{M}, [\![\,]\!])$.*

PROOF. (i) The α-axioms of $\lambda\mathsf{Y}\eta(\mathcal{A})$ are the following equations:

(1) $\mathbf{R}_i\mathbf{c}_{\vec{a}} = \mathbf{true}_0$ for $\vec{a} \in R_i$

(2) $\mathbf{R}_i\mathbf{c}_{\vec{a}} = \mathbf{false}_0$ for $\vec{a} \in A^{p_i} - R_i$

(3) $\mathbf{F}_j\mathbf{c}_{\vec{a}} = \mathbf{c}_{f_j(\vec{a})}$ for $\vec{a} \in A^{q_j}$.

Let $\vec{a} \in R_i$. Then for all $b, c \in D$

$$\begin{aligned} [\![\mathbf{R}_i\mathbf{c}_{\vec{a}}]\!]bc &= R_i^* a_1^* \ldots a_{p_i}^* bc \\ &= \varphi_i(a_1^*, \ldots, a_{p_i}^*, b, c) \\ &= b \\ &= [\![\mathbf{true}_0]\!]bc. \end{aligned}$$

So $[\![\mathbf{R}_i\mathbf{c}_{\vec{a}}]\!] \approx [\![\mathbf{true}_0]\!]$. But $\mathfrak{M}$ is extensional, so $[\![\mathbf{R}_i\mathbf{c}_{\vec{a}}]\!] = [\![\mathbf{true}_0]\!]$ and therefore $\mathfrak{M}, [\![\ ]\!] \models \mathbf{R}_i\mathbf{c}_{\vec{a}} = \mathbf{true}_0$.

Satisfaction of $(\mathbf{R}_i\mathbf{c}_{\vec{a}} = \mathbf{false}_0)$ for $\vec{a} \in A^{p_i} - R_i$ is shown similarly.

Now let $\vec{a} \in A^{q_j}$. Then

$$\begin{aligned}
[\![\mathbf{F}_j\mathbf{c}_{\vec{a}}]\!] &= f_j^* a_1^* \dots a_{q_j}^* \\
&= \psi_j(a_1^*, \dots, a_{q_j}^*) \\
&= f_j(\vec{a})^* \\
&= [\![\mathbf{c}_{f_j(\vec{a})}]\!].
\end{aligned}$$

So $[\![\mathbf{F}_j\mathbf{c}_{\vec{a}}]\!] = [\![\mathbf{c}_{f_j(\vec{a})}]\!]$ and therefore $\mathfrak{M}, [\![\,]\!] \models \mathbf{F}_j\mathbf{c}\vec{a} = \mathbf{c}_{f_j(\vec{a})}$.

(ii) The satisfaction of the γ-axioms is shown as in the proof of lemma 3.5.7. □

Because $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ is extensional it follows that $(\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D}), [\![\,]\!])$ is a $\lambda\mathsf{Y}\eta(\mathcal{A})$-model. This model is called a monotone $\lambda\mathsf{Y}\eta(\mathcal{A})$-model. We now have the following theorem.

3.5.11. Theorem. *If $\mathcal{A}$ is $\mathbb{D}$-representable, then $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ is a pre-$\lambda\mathsf{Y}\eta(\mathcal{A})$-model.* □

Recall that $\mathbb{A}_\perp$ is the flat cpo over A. Next it will be shown that the structure $\mathcal{A}$ is $\mathbb{A}_\perp$-representable. Let ${}^* : A \to A_\perp$ be the inclusion mapping, so that $a^* = a$ for all $a \in A$.

3.5.12. Definition. Let $m \in \omega$.

(i) Let $R \subseteq A^m$. Define $\Delta_R : (A_\perp)^{m+2} \to A_\perp$ by

(1) $\Delta_R(\vec{a}, b, c) = b$ if $\vec{a} \in R$

(2) $\Delta_R(\vec{a}, b, c) = c$ if $\vec{a} \in A^m - R$

(3) $\Delta_R(\vec{a}, b, c) = \perp$ if $\vec{a} \notin A^m$.

(ii) Let $f : A^m \to A$. Define $\bar{f} : (A_\perp)^m \to A_\perp$ by

(1) $\bar{f}(\vec{a}) = f(\vec{a})$ if $\vec{a} \in A^m$

(2) $\bar{f}(\vec{a}) = \perp$ if $\vec{a} \notin A^m$. □

It is easy to see that for $R \subseteq A^m$ the function Δ_R is monotone and represents R w.r.t. *, and it is also easy to see that for $f : A^m \to A$ the function $\bar{f}$ is monotone and represents f w.r.t. *. So every relation on A is $\mathbb{A}_\perp$-representable w.r.t. * and every function on A is $\mathbb{A}_\perp$-representable w.r.t. *. This gives us the following lemma.

3.5.13. Lemma. *The structure $\mathcal{A}$ is $\mathbb{A}_\perp$-representable.* □

3.5.14. Theorem. *$\mathfrak{M}_{\mathbf{Mon}}(\mathbb{A}_\perp)$ is a pre-$\lambda\mathsf{Y}\eta(\mathcal{A})$-model.* □

Let $[\![\,]\!]$ be the $\lambda\mathsf{Y}\eta(\mathcal{A})$-interpretation in $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{A}_\perp)$ such that

- $[\![\mathbf{Y}_\sigma]\!] = \mathbb{Y}_\sigma$ for every $\sigma \in \mathbb{T}$
- $[\![\mathbf{c}_a]\!] = a$ for all $a \in A$
- $[\![\mathbf{R}_i]\!]$ $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{A}_\perp)$-represents Δ_{R_i} for every $i \in \{1, \ldots, r\}$
- $[\![\mathbf{F}_j]\!]$ $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{A}_\perp)$-represents $\bar{f}_j$ for every $j \in \{1, \ldots, s\}$.

By the proof of theorem 3.5.11 it follows that $(\mathfrak{M}_{\mathbf{Mon}}(\mathbb{A}_\perp), [\![\,]\!])$ is a $\lambda\mathsf{Y}\eta(\mathcal{A})$-model. This model is called the standard monotone $\lambda\mathsf{Y}\eta(\mathcal{A})$-model. The interpretation $[\![\,]\!]$ is called the standard $\lambda\mathsf{Y}\eta(\mathcal{A})$-interpretation.

3.5.15. Definition. The $\lambda\mathsf{Y}(\mathcal{A})$-definable objects of $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{A}_\perp)$ are the definable objects of the standard monotone $\lambda\mathsf{Y}\eta(\mathcal{A})$-model. $\square$

Let $\omega_\perp$ be the flat cpo over ω. The type structure $\mathfrak{M}_{\mathbf{Mon}}(\omega_\perp)$ will be denoted by **MON**. By the previous theorem we obtain the following theorem.

3.5.16. Theorem. *The type structure* **MON** *is a pre-$\lambda\mathsf{Y}\eta(\mathcal{N})$-model.* $\square$

Using theorem 3.5.11 we can prove non-computability results for some systems $\lambda\mathsf{Y}\eta(\mathcal{A})$. As an example of this we show the non-computability of the predecessor function P^- in the system $\lambda\mathsf{Y}\eta(\mathcal{A})$ with $\mathcal{A} = (\omega, \{0\}, S^+)$. So $\lambda\mathsf{Y}\eta(\mathcal{A})$ only contains test-for-zero and successor on the natural numbers, and not equality as in $\lambda\mathsf{Y}\eta(\mathcal{N})$.

3.5.17. Theorem. *Let $\mathcal{A} = (\omega, \{0\}, S^+)$. Then the predecessor function P^- is not $\lambda\mathsf{Y}\eta(\mathcal{A})$-computable.*

Proof. Let ∞ be some object with $\infty \notin \omega_\perp$, and define $D = \omega_\perp \cup \{\infty\}$. Let $\sqsubseteq$ be the least partial ordering on D such that
(i) $\perp \sqsubseteq 0$
(ii) $\perp \sqsubseteq 1 \sqsubseteq 2 \sqsubseteq 3 \sqsubseteq \ldots \sqsubseteq \infty$.
Then $0 \not\sqsubseteq 1$ and $\mathbb{D} = (D, \sqsubseteq)$ is a cpo. It can easily be shown that $\mathcal{A}$ is $\mathbb{D}$-representable w.r.t. the inclusion mapping $\omega \to D$. So there exists a $\lambda\mathsf{Y}\eta(\mathcal{A})$-interpretation $[\![\,]\!]$ in $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D})$ such that $(\mathfrak{M}_{\mathbf{Mon}}(\mathbb{D}), [\![\,]\!])$ is a $\lambda\mathsf{Y}\eta(\mathcal{A})$-model and $[\![\mathbf{c}_m]\!] = m$ for all $m \in \omega$.

Now suppose P^- is $\lambda\mathsf{Y}\eta(\mathcal{A})$-computable. Let $F \in \mathcal{T}_{0\to 0}(\lambda\mathsf{Y}\eta(\mathcal{A}))^0$ be such that F $\lambda\mathsf{Y}\eta(\mathcal{A})$-computes P^- w.r.t. $\mathbf{c}$. Define $\chi = [\![F]\!]$. Then $\chi \in D_{0\to 0}$, so χ is a monotone function from D into D with for all $m \in \omega$

$$\begin{aligned}
\chi(m) &= [\![F]\!]m \\
&= [\![F]\!][\![\mathbf{c}_m]\!] \\
&= [\![F\mathbf{c}_m]\!] \\
&= [\![\mathbf{c}_{P^-(m)}]\!] \\
&= P^-(m).
\end{aligned}$$

Note that in $\mathbb{D}$ we have $1 \sqsubseteq 2$. But then, by the monotonicity of χ we have

$$0 = P^-(1) = \chi(1) \sqsubseteq \chi(2) = P^-(2) = 1$$

so $0 \sqsubseteq 1$. Contradiction. So P^- is not $\lambda\mathsf{Y}\eta(\mathcal{A})$-computable. $\square$

In the remainder of this section we will prove some new results about the type structure $\mathbf{Tot}(A)$ consisting of all so-called total objects of $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{A}_\perp)$.

In the following $\mathbb{D}_0 = \mathbb{A}_\perp$ and $\mathbb{D}_{\sigma\to\tau} = \mathbf{Mon}(\mathbb{D}_\sigma, \mathbb{D}_\tau)$. For every $\sigma \in \mathbb{T}$ let $\mathbb{D}_\sigma = (D_\sigma, \sqsubseteq_\sigma)$.

3.5.18. Definition. (i) For every $\sigma \in \mathbb{T}$ define $\mathbf{Tot}_\sigma(A) \subseteq D_\sigma$ by induction on the structure of σ as follows.

(1) $\mathbf{Tot}_0(A) = A$

(2) $\mathbf{Tot}_{\sigma\to\tau}(A) = \{a \in D_{\sigma\to\tau} \mid \forall b \in \mathbf{Tot}_\sigma(A)[\, ab \in \mathbf{Tot}_\tau(A)\,]\,\}$.

(ii) The substructure $\mathbf{Tot}(A)$ of $\mathfrak{M}_{\mathbf{Mon}}(A)$ is defined by

$$\mathbf{Tot}(A) = ((\mathbf{Tot}_\sigma(A))_{\sigma\in\mathbb{T}}, (app_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}})$$

where $app_{\sigma,\tau} : \mathbf{Tot}_{\sigma\to\tau}(A) \times \mathbf{Tot}_\sigma(A) \to \mathbf{Tot}_\tau(A)$ is defined by $app_{\sigma,\tau}(a,b) = ab$.

(iii) The objects of $\mathbf{Tot}(A)$ are called the total objects of $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{A}_\perp)$.

(iv) Extensional equality in $\mathbf{Tot}(A)$ is denoted by $\sim$, so for all $\sigma \in \mathbb{T}$ and $a, b \in \mathbf{Tot}_\sigma(A)$

$$a \sim b \iff \forall \vec{c} \in (\mathbf{Tot}(A))_\sigma[\, a\vec{c} = b\vec{c}\,].\ \square$$

For all $\sigma \in \mathbb{T}$ and $a \in D_\sigma$ we have

$$a \in \mathbf{Tot}_\sigma(A) \iff \forall \vec{b} \in (\mathbf{Tot}(A))_\sigma[\, a\vec{b} \in A\,].$$

Every element of A is total. For every $\sigma \in \mathbb{T}$ let $\perp^\sigma$ be the least element of $\mathbb{D}_\sigma$. Then for all $\sigma \in \mathbb{T}$ and $\vec{a} \in (\mathfrak{M}_{\mathbf{Mon}}(\mathbb{A}_\perp))_\sigma$ we have $\perp^\sigma \vec{a} = \perp$, so $\perp^\sigma$ is not total.

3.5.19. Lemma. *Let $\sigma \in \mathbb{T}$, $a, b \in D_\sigma$.*

(i) $a \in \mathbf{Tot}_\sigma(A)\ \&\ a \sqsubseteq b \Rightarrow b \in \mathbf{Tot}_\sigma(A)$.

(ii) $a, b \in \mathbf{Tot}_\sigma(A)\ \&\ a \sqsubseteq b \Rightarrow a \sim b$.

Proof. (i) Let $a \in \mathbf{Tot}_\sigma(A)$ and $a \sqsubseteq b$. Let $\vec{c} \in (\mathbf{Tot}(A))_\sigma$. Then $a\vec{c} \in A$ and $a\vec{c} \sqsubseteq b\vec{c}$. So $b\vec{c} \in A$. Therefore $b \in \mathbf{Tot}_\sigma(A)$.

(ii) Let $a, b \in \mathbf{Tot}_\sigma(A)$ and $a \sqsubseteq b$. Let $\vec{c} \in (\mathbf{Tot}(A))_\sigma$. Then $a\vec{c} \in A$ and $a\vec{c} \sqsubseteq b\vec{c}$. So $a\vec{c} = b\vec{c}$. Therefore $a \sim b$. $\square$

For every $a \in A$ we define two total functions C_a and S_a from $A_\perp$ into $A_\perp$ as follows. First define $C_a : A_\perp \to A_\perp$ by $C_a(p) = a$. Then C_a is the constant-a function. Define $S_a : A_\perp \to A_\perp$ by

- $S_a(p) = a$ if $p \in A$
- $S_a(p) = \perp$ if $p = \perp$.

Then S_a is the strict constant-a function. For all $a \in A$ we have that C_a and S_a are monotone, so $C_a \in D_{0\to 0}$ and $S_a \in D_{0\to 0}$, and C_a and S_a are total, and therefore $C_a \in \mathbf{Tot}_{0\to 0}(A)$ and $S_a \in \mathbf{Tot}_{0\to 0}(A)$. For all $a \in A$ the functions C_a and S_a have the same behaviour on A, so $C_a \sim S_a$, that is C_a and S_a are extensionally equal in $\mathbf{Tot}(A)$. But $C_a \neq S_a$, so $\mathbf{Tot}(A)$ is not extensional. However, $\mathbf{Tot}(A)$ is quasi-extensional, as will be shown in a few moments.

First we are going to embed the full type structure $\mathfrak{M}(A)$ into $\mathfrak{M}_{\mathbf{Mon}}(A)$. Recall that $A_0 = A$ and $A_{\sigma\to\tau} = (A_\tau)^{A_\sigma}$. For all $\sigma \in \mathbb{T}$ and $a, b \in D_\sigma$ let

$$a \uparrow b \Longleftrightarrow \exists c \in D_\sigma[\, a \sqsubseteq c \,\&\, b \sqsubseteq c\,].$$

In the following we define, for every $\sigma \in \mathbb{T}$ a function $h_\sigma : A_\sigma \to D_\sigma$ such that

$$\forall p, q \in A_\sigma[\, h_\sigma(p) \uparrow h_\sigma(q) \Rightarrow p = q\,].$$

This will be done by induction on the structure of σ. The embedding will then be $h = (h_\sigma)_{\sigma\in\mathbb{T}}$.

1. Define $h_0 : A_0 \to D_0$ by $h_0(p) = p$ for all $p \in A_0$. Then for all $p, q \in A_0$

$$h_0(p) \uparrow h_0(q) \Rightarrow p = q.$$

2. Let $\sigma, \tau \in \mathbb{T}$, $h_\sigma : A_\sigma \to D_\sigma$ and $h_\tau : A_\tau \to D_\tau$ be such that

$$\forall p, q \in A_\sigma[\, h_\sigma(p) \uparrow h_\sigma(q) \Rightarrow p = q\,]$$

$$\forall p, q \in A_\tau[\, h_\tau(p) \uparrow h_\tau(q) \Rightarrow p = q\,].$$

   We define $h_{\sigma\to\tau} : A_{\sigma\to\tau} \to D_{\sigma\to\tau}$ as follows. Let $f \in A_{\sigma\to\tau}$. Then $f : A_\sigma \to A_\tau$. Define $h_{\sigma\to\tau}(f) : D_\sigma \to D_\tau$ as follows. Let $a \in D_\sigma$ and distinguish two cases.

   - There exists $p \in A_\sigma$ with $h_\sigma(p) \sqsubseteq a$. Then, by the induction hypothesis for σ, this p is unique. Define

$$h_{\sigma\to\tau}(f)(a) = h_\tau(f(p)).$$

   - There is no $p \in A_\sigma$ with $h_\sigma(p) \sqsubseteq a$. Then define

$$h_{\sigma\to\tau}(f)(a) = \perp^\tau.$$

First we show that $h_{\sigma\to\tau}(f)$ is monotone, so that $h_{\sigma\to\tau}(f) \in D_{\sigma\to\tau}$. Let $a, b \in D_\sigma$ with $a \sqsubseteq b$. Distinguish two cases.

- There exists $p \in A_\sigma$ with $h_\sigma(p) \sqsubseteq a$. Then $h_\sigma(p) \sqsubseteq b$ and so
$$h_{\sigma\to\tau}(f)(a) = h_\tau(f(p)) = h_{\sigma\to\tau}(f)(b).$$
- There is no $p \in A_\sigma$ with $h_\sigma(p) \sqsubseteq a$. Then
$$h_{\sigma\to\tau}(f)(a) = \bot^\tau \sqsubseteq h_{\sigma\to\tau}(f)(b).$$

So, in any case $h_{\sigma\to\tau}(f)(a) \sqsubseteq h_{\sigma\to\tau}(f)(b)$. So $h_{\sigma\to\tau}(f)$ is monotone and $h_{\sigma\to\tau}(f) \in D_{\sigma\to\tau}$.

Next we verify the induction hypothesis for type $\sigma \to \tau$. Let $f, g \in A_{\sigma\to\tau}$ be such that $h_{\sigma\to\tau}(f) \uparrow h_{\sigma\to\tau}(g)$. Let $a \in D_{\sigma\to\tau}$ be such that $h_{\sigma\to\tau}(f) \sqsubseteq a$ and $h_{\sigma\to\tau}(g) \sqsubseteq a$. We show $f = g$.

Let $p \in A_\sigma$. Then $h_\sigma(p) \in D_\sigma$ and $h_\sigma(p) \sqsubseteq h_\sigma(p)$, so by the definition of $h_{\sigma\to\tau}$
$$h_{\sigma\to\tau}(f)(h_\sigma(p)) = h_\tau(f(p))$$
$$h_{\sigma\to\tau}(g)(h_\sigma(p)) = h_\tau(g(p)).$$
Because $h_{\sigma\to\tau}(f) \sqsubseteq a$ we have $h_\tau(f(p)) \sqsubseteq a \cdot h_\sigma(p)$ and because $h_{\sigma\to\tau}(g) \sqsubseteq a$ we have $h_\tau(g(p)) \sqsubseteq a \cdot h_\sigma(p)$. So $h_\tau(f(p)) \uparrow h_\tau(g(p))$. Therefore, by the induction hypothesis for τ we get $f(p) = g(p)$. So $f = g$. Therefore
$$\forall f, g \in A_{\sigma\to\tau}[\, h_{\sigma\to\tau}(f) \uparrow h_{\sigma\to\tau}(g) \Rightarrow f = g\,].$$

We have now defined $h_\sigma : A_\sigma \to D_\sigma$ for every $\sigma \in \mathbb{T}$, with the property
$$\forall p, q \in A_\sigma[\, h_\sigma(p) \uparrow h_\sigma(q) \Rightarrow p = q\,].$$
Then every h_σ is injective, because for all $p, q \in A_\sigma$
$$h_\sigma(p) = h_\sigma(q) \Rightarrow h_\sigma(p) \uparrow h_\sigma(q) \Rightarrow p = q.$$
Now define $h = (h_\sigma)_{\sigma\in\mathbb{T}}$. We show that h is a homomorphism from $\mathfrak{M}(A)$ into $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{A}_\bot)$ as follows.

Let $\sigma, \tau \in \mathbb{T}$, $p \in A_{\sigma\to\tau}$, $q \in A_\tau$. Then $h_{\sigma\to\tau}(p) : D_\sigma \to D_\tau$ and $h_\sigma(q) \in D_\sigma$ with $h_\sigma(q) \sqsubseteq h_\sigma(q)$. Therefore $h_{\sigma\to\tau}(p)h_\sigma(q) = h_{\sigma\to\tau}(p)(h_\sigma(q)) = h_\tau(p(q)) = h_\tau(pq)$.

So h is an embedding from $\mathfrak{M}(A)$ into $\mathfrak{M}_{\mathbf{Mon}}(\mathbb{A}_\bot)$ with the property that for all $\sigma \in \mathbb{T}$ and $p, q \in A_\sigma$
$$h_\sigma(p) \uparrow h_\sigma(q) \Rightarrow p = q.$$

3.5.20. LEMMA. *For all* $\sigma \in \mathbb{T}$
(i) $\forall p \in A_\sigma [\, h_\sigma(p) \in \mathbf{Tot}_\sigma(A)\,]$
(ii) $\forall a \in \mathbf{Tot}_\sigma(A) \exists p \in A_\sigma [\, h_\sigma(p) \sqsubseteq a\,]$.

PROOF. By induction on the structure of σ. For $\sigma = 0$ the proof is simple. Let $\sigma, \tau \in \mathbb{T}$.

(i) Let $p \in A_{\sigma\to\tau}$. Let $b \in \mathbf{Tot}_\sigma(A)$. By induction there exists $q \in A_\sigma$ with $h_\sigma(q) \sqsubseteq b$. Then $h_{\sigma\to\tau}(p)b = h_\tau(pq)$. By induction $h_\tau(pq) \in \mathbf{Tot}_\tau(A)$, so $h_{\sigma\to\tau}(p)b \in \mathbf{Tot}_\tau(A)$. Therefore $h_{\sigma\to\tau}(p) \in \mathbf{Tot}_{\sigma\to\tau}(A)$.

(ii) Let $a \in \mathbf{Tot}_{\sigma\to\tau}(A)$. We will show that

$$\forall p \in A_\sigma \exists! q \in A_\tau [\, h_\tau(q) \sqsubseteq a \cdot h_\sigma(p)\,].$$

Let $p \in A_\sigma$. Then $h_\sigma(p) \in \mathbf{Tot}_\sigma(A)$ by induction. So $a \cdot h_\sigma(p) \in \mathbf{Tot}_\tau(A)$. Again by induction there exists $q \in A_\tau$ with $h_\tau(q) \sqsubseteq a \cdot h_\sigma(p)$. Note that this q is unique.

So there exists $f \in A_{\sigma\to\tau}$ with $h_\tau(fp) \sqsubseteq a \cdot h_\sigma(p)$ for all $p \in A_\sigma$. We will show that $h_{\sigma\to\tau}(f) \sqsubseteq a$.

Let $b \in D_\sigma$ and distinguish two cases.

- There exists $p \in A_\sigma$ with $h_\sigma(p) \sqsubseteq b$. Then $h_{\sigma\to\tau}(f)b = h_\tau(fp) \sqsubseteq a \cdot h_\sigma(p) \sqsubseteq ab$.
- There is no $p \in A_\sigma$ with $h_\sigma(p) \sqsubseteq b$. Then $h_{\sigma\to\tau}(f)b = \bot^\tau \sqsubseteq ab$.

In any case $h_{\sigma\to\tau}(f)b \sqsubseteq ab$. Therefore $h_{\sigma\to\tau}(f) \sqsubseteq a$. $\square$

By (i) of this lemma we see that h is an embedding of $\mathfrak{M}(A)$ in $\mathbf{Tot}(A)$.

3.5.21. LEMMA. *For every* $\sigma \in \mathbb{T}$

$$\forall a \in \mathbf{Tot}_\sigma(A) \exists p \in A_\sigma [\, h_\sigma(p) \sim a\,].$$

PROOF. Let $\sigma \in \mathbb{T}$ and $a \in \mathbf{Tot}_\sigma(A)$. Then there exists $p \in A_\sigma$ with $h_\sigma(p) \sqsubseteq a$. Then $h_\sigma(p) \in \mathbf{Tot}_\sigma(A)$ and so $h_\sigma(p) \sim a$. $\square$

3.5.22. LEMMA. *For all* $\sigma \in \mathbb{T}$, $p \in A_\sigma$ *and* $a \in D_\sigma$

$$h_\sigma(p) \sim a \iff h_\sigma(p) \sqsubseteq a.$$

PROOF. Note that $(\Leftarrow)$ follows by lemma 3.5.19. We prove $(\Rightarrow)$ by induction on the structure of σ as follows. For $\sigma = 0$ the proof is simple. Let $\sigma, \tau \in \mathbb{T}$. Let $p \in A_{\sigma\to\tau}$ and $a \in D_{\sigma\to\tau}$ with $h_{\sigma\to\tau}(p) \sim a$. We have to show that $h_{\sigma\to\tau}(p) \sqsubseteq a$. Let $b \in D_\sigma$ and distinguish two cases.

- There exists $q \in A_\sigma$ with $h_\sigma(q) \sqsubseteq b$. Then $h_{\sigma\to\tau}(p)b = h_\tau(pq)$. Also $b \in \mathbf{Tot}_\sigma(A)$ because $h_\sigma(q) \in \mathbf{Tot}_\sigma(A)$. So $h_{\sigma\to\tau}(p)b \sim ab$ because $h_{\sigma\to\tau}(p) \sim a$. So $h_\tau(pq) \sim ab$. By induction $h_\tau(pq) \sqsubseteq ab$ and so $h_{\sigma\to\tau}(p)b \sqsubseteq ab$.

- There is no $q \in A_\sigma$ with $h_\sigma(q) \sqsubseteq b$. Then $h_{\sigma\to\tau}(p)b = \bot^\tau \sqsubseteq ab$.

In any case $h_{\sigma\to\tau}(p)b \sqsubseteq ab$. Therefore $h_{\sigma\to\tau}(p) \sqsubseteq a$. □

3.5.23. Lemma. *For all $\sigma \in \mathbb{T}$ and $p, q \in A_\sigma$*

$$h_\sigma(p) \sim h_\sigma(q) \Rightarrow p = q.$$

Proof. Let $\sigma \in \mathbb{T}$, $p, q \in A_\sigma$ with $h_\sigma(p) \sim h_\sigma(q)$. Then $h_\sigma(p) \sqsubseteq h_\sigma(q)$ by the previous lemma, so $h_\sigma(p) \uparrow h_\sigma(q)$ and therefore $p = q$. □

3.5.24. Lemma. *For all $\sigma, \tau \in \mathbb{T}$, $p \in A_{\sigma\to\tau}$, $q \in A_\sigma$, $a \in D_{\sigma\to\tau}$ and $b \in D_\sigma$*

$$h_{\sigma\to\tau}(p) \sim a \;\&\; h_\sigma(q) \sim b \Rightarrow h_\tau(pq) \sim ab.$$

Proof. Let $\sigma, \tau \in \mathbb{T}$, $p \in A_{\sigma\to\tau}$, $q \in A_\sigma$, $a \in D_{\sigma\to\tau}$, $b \in D_\sigma$ with $h_{\sigma\to\tau}(p) \sim a$ and $h_\sigma(q) \sim b$. Then $h_\sigma(q) \sqsubseteq b$. So $h_{\sigma\to\tau}(p)b = h_\tau(pq)$. Also $h_{\sigma\to\tau}(p)b \sim ab$. So $h_\tau(pq) \sim ab$. □

3.5.25. Theorem. $\mathbf{Tot}(A)$ *is quasi-extensional.*

Proof. Let $\sigma, \tau \in \mathbb{T}$, $a, a' \in \mathbf{Tot}_{\sigma\to\tau}$, $b, b' \in \mathbf{Tot}_\sigma(A)$ with $a \sim a'$ and $b \sim b'$. There exist $p \in A_{\sigma\to\tau}$ and $q \in A_\sigma$ with $h_{\sigma\to\tau}(p) \sim a$ and $h_\sigma(q) \sim b$. Then also $h_{\sigma\to\tau}(p) \sim a'$ and $h_\sigma(q) \sim b'$, and so $ab \sim h_\tau(pq) \sim a'b'$. Therefore $\mathbf{Tot}(A)$ is quasi-extensional. □

3.5.26. Theorem. $\mathfrak{M}(A) \cong \mathbf{Tot}(A)^E$.

Proof. Let $\pi : \mathbf{Tot}(A) \to \mathbf{Tot}(A)^E$ be the quotient mapping $a \mapsto [a] = \{b \in \mathbf{Tot}_\sigma \mid a \sim b\}$ for all $a \in \mathbf{Tot}_\sigma(A)$. Define $f = \pi \circ h$. Then $f : \mathfrak{M}(A) \to \mathbf{Tot}(A)^E$.

(i) Let $\sigma \in \mathbb{T}$, $p, q \in A_\sigma$ with $f(p) = f(q)$. Then $[h_\sigma(p)] = [h_\sigma(q)]$, so $h_\sigma(p) \sim h_\sigma(q)$, so $p = q$. So f is injective.

(ii) Let $\sigma \in \mathbb{T}$ and $[a] \in \mathbf{Tot}_\sigma(A)^E$. Then $a \in \mathbf{Tot}_\sigma(A)$ so there exists $p \in A_\sigma$ with $h_\sigma(p) \sim a$. Then $[a] = [h_\sigma(p)] = f(p)$. So f is surjective.

By (i) and (ii) it follows that f is bijective. Therefore f is an isomorphism and $\mathfrak{M}(A) \cong \mathbf{Tot}(A)^E$. □

Chapter 4

Enumerators

In this chapter we study enumerators in simply typed λ-calculi. In section 4.1 we introduce the system $\lambda\mathsf{E}(\mathcal{N})$ which contains enumerators, and we study its basic properties. The remainder of this chapter is devoted to the search for $\lambda\mathsf{E}(\mathcal{N})$-models. Our main tool in this search is the notion of recursion theory on a type structure, which is introduced in section 4.3. In section 4.4 it is shown that a quasi-extensional type structure carries a recursion theory if and only if it is a pre-quasi-$\lambda\mathsf{E}(\mathcal{N})$-model. In section 4.5 a recursion theory is constructed on the type structure **MON**, thereby proving that **MON** is a pre-$\lambda\mathsf{E}(\mathcal{N})$-model.

4.1. Simply typed λ-calculi with enumerators.

4.1.1. DEFINITION. Let Σ be a system and let $\mathbf{a}$ be a numeral system in Σ. Then Σ has enumerators w.r.t. $\mathbf{a}$ if for every $\sigma \in \mathbb{T}$ there exists $\mathsf{E}_\sigma \in \mathcal{T}_{0\to\sigma}(\Sigma)^0$ such that for every $M \in \mathcal{T}_\sigma(\Sigma)^0$ there exists $m \in \omega$ such that

$$\Sigma \vdash \mathsf{E}_\sigma \mathbf{a}_m = M.$$

The term E_σ is called an enumerator for type σ. $\square$

First it will be shown that every β-system that has enumerators, also has fixed point combinators.

4.1.2. THEOREM. *Let Σ be a β-system and let $\mathbf{a}$ be a numeral system in Σ such that Σ has enumerators w.r.t. $\mathbf{a}$. Then Σ has fixed point combinators, that is for every $\sigma \in \mathbb{T}$ there exists $\mathsf{Y}_\sigma \in \mathcal{T}_{(\sigma\to\sigma)\to\sigma}(\Sigma)^0$ such that*

$$\Sigma \vdash \mathsf{Y}_\sigma = \lambda f.f(\mathsf{Y}_\sigma f).$$

PROOF. Let $\sigma \in \mathbb{T}$. Define $\tau = 0 \to (\sigma \to \sigma) \to \sigma$ and $A_\sigma \equiv \lambda af.f(\mathsf{E}_\tau aaf)$ (with $a \in Var_0$ and $f \in Var_{\sigma\to\sigma}$). Then $A_\sigma \in \mathcal{T}_\tau(\Sigma)^0$, so there exists $m \in \omega$ with

$\Sigma \vdash \mathsf{E}_\tau \mathbf{a}_m = A_\sigma$. Now define $\mathsf{Y}_\sigma \equiv A_\sigma \mathbf{a}_m$. Then, in Σ, we have

$$\begin{aligned} \mathsf{Y}_\sigma &= (\lambda af.f(\mathsf{E}_\tau aaf))\mathbf{a}_m \\ &= \lambda f.f(\mathsf{E}_\tau \mathbf{a}_m \mathbf{a}_m f) \\ &= \lambda f.f(A_\sigma \mathbf{a}_m f) \\ &= \lambda f.f(\mathsf{Y}_\sigma f). \ \Box \end{aligned}$$

We see that fixed point combinators can be obtained from enumerators. Next it will be shown that the systems $\lambda\mathsf{Y}(\mathcal{N})$ and $\lambda\mathsf{Y}\eta(\mathcal{N})$ do not have enumerators w.r.t. $\mathbf{c}$. It follows that enumerators cannot be constructed from fixed point combinators.

4.1.3. Definition. Let $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0 \in \mathbb{T}$.
(i) $\mathbf{\Omega}_\sigma \equiv \mathbf{Y}_\sigma \mathbf{I}_\sigma \in \mathcal{T}_\sigma(\lambda\mathsf{Y}(\mathcal{N}))^0$.
(ii) For every $m \in \omega$ define

$$\mathbf{\Gamma}^\sigma_m \equiv \lambda\vec{x}.\mathbf{c}_m \in \mathcal{T}_\sigma(\lambda\mathsf{Y}(\mathcal{N}))^0.$$

with $\vec{x} = x_1, \ldots, x_k$ and $x_i \in Var_{\sigma_i}$ $(i = 1, \ldots, k)$.
(iii) $\mathbf{Z}_\sigma \equiv \mathbf{\Omega}_\sigma \mathbf{\Gamma}^{\sigma_1}_0 \ldots \mathbf{\Gamma}^{\sigma_k}_0$. $\Box$

Note that for every $\sigma \in \mathbb{T}$ $\mathbf{Z}_\sigma$ is a closed $\lambda\mathsf{Y}(\mathcal{N})$-term of type 0.

4.1.4. Lemma. *Let $\sigma \in \mathbb{T}$.*
(i) $\mathbf{\Omega}_\sigma \twoheadrightarrow_{\alpha\beta\gamma\eta} M \Rightarrow \mathbf{Y}_\sigma \in M$.
(ii) $\lambda\vec{x}.\mathbf{Z}_\sigma \twoheadrightarrow_{\alpha\beta\gamma\eta} M \Rightarrow \mathbf{Y}_\sigma \in M$.

Proof. (i) Let $\mathbf{\Omega}_\sigma \twoheadrightarrow_{\alpha\beta\gamma\eta} M$. Then $\mathbf{\Omega}_\sigma \twoheadrightarrow_{\beta\gamma\eta} M$. The only way to loose $\mathbf{Y}_\sigma$ in a $\beta\gamma\eta$-reduction is by contracting a β-redex $(\lambda x.P)Q$ with $x \notin FV(P)$ and $\mathbf{Y}_\sigma \in Q$. But this never happens in a reduction starting with $\mathbf{\Omega}_\sigma$ because every abstraction term that occurs as a subterm of a reduct of $\mathbf{\Omega}_\sigma$ is of the form $\lambda x.P$ with $x \in FV(P)$. Therefore $\mathbf{Y}_\sigma \in M$.

(ii) Let $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0$ and suppose $\lambda\vec{x}.\mathbf{Z}_\sigma \twoheadrightarrow_{\alpha\beta\gamma\eta} M$. Then $M \equiv \lambda\vec{x}.U\mathbf{\Gamma}^{\sigma_1}_0 \ldots \mathbf{\Gamma}^{\sigma_k}_0$ with $\mathbf{\Omega}_\sigma \twoheadrightarrow_{\alpha\beta\gamma\eta} U$. Then by (i) $\mathbf{Y}_\sigma \in U$ and so $\mathbf{Y}_\sigma \in M$. $\Box$

4.1.5. Proposition. *Let $\sigma \in \mathbb{T}$. Then there does not exist a term*

$$\mathsf{E}_\sigma \in \mathcal{T}_{0\to\sigma}(\lambda\mathsf{Y}(\mathcal{N}))$$

such that for every $M \in \mathcal{T}_\sigma(\lambda\mathsf{Y}(\mathcal{N}))^0$ there exists $m \in \omega$ with

$$\mathsf{E}_\sigma \mathbf{c}_m =_{\alpha\beta\gamma\eta} M.$$

PROOF. Suppose such E_σ exists. Let $\tau \in \mathbb{T}$ be such that $\mathbf{Y}_\tau \notin \mathsf{E}_\sigma$. Take $M \equiv \lambda\vec{x}.\mathbf{Z}_\tau$ with $\vec{x}$ such that $M \in \mathcal{T}_\sigma(\lambda\mathsf{Y}(\mathcal{N}))^0$. Then there exists $m \in \omega$ with $\mathsf{E}_\sigma\mathbf{c}_m =_{\alpha\beta\gamma\eta} M$. By $\mathbf{CR}(\alpha\beta\gamma\eta)$ there exists $N \in \mathcal{T}_\sigma(\lambda\mathsf{Y}(\mathcal{N}))$ such that $\mathsf{E}_\sigma\mathbf{c}_m \twoheadrightarrow_{\alpha\beta\gamma\eta} N$ and $M \twoheadrightarrow_{\alpha\beta\gamma\eta} N$. By the previous lemma it follows that $\mathbf{Y}_\tau \in N$. But since $\alpha\beta\gamma\eta$-reduction does not introduce new $\mathbf{Y}$-constants, we have that $\mathbf{Y}_\tau \in (\mathsf{E}_\sigma\mathbf{c}_m)$ and so $\mathbf{Y}_\tau \in \mathsf{E}_\sigma$. Contradiction. □

4.1.6. THEOREM. *The systems $\lambda\mathsf{Y}(\mathcal{N})$ and $\lambda\mathsf{Y}\eta(\mathcal{N})$ do not have enumerators w.r.t.* $\mathbf{c}$.

PROOF. By 4.1.5. □

Using δ-reduction we can build systems with enumerators. The δ-reduction that introduces enumerators will be called ε-reduction.

Recall that $\omega^* = \bigcup\{\omega^k |\ k \in \omega\}$ is the set of all finite sequences of natural numbers. In the following

$$< >: \omega^* \rightarrowtail \omega$$

$$(m_0, \ldots, m_{k-1}) \mapsto < m_0, \ldots, m_{k-1} >$$

is an injective coding of the finite sequences such that for every $k \in \omega$ the restriction of $< >$ to ω^k is a recursive function $\omega^k \to \omega$. For every $i \in \omega$ let

$$(\ )_i : \omega \to \omega$$

$$m \mapsto (m)_i$$

be a recursive decoding function such that for all $k, i \in \omega$ with $i < k$ and all $m_0, \ldots, m_{k-1} \in \omega$

$$(< m_0, \ldots, m_{k-1} >)_i = m_i.$$

4.1.7. DEFINITION. Let $\# : \mathbb{T} \to \omega$ be defined by induction on the structure of the types as follows.

(i) $\#(0) =< 0 >$.

(ii) $\#(\sigma \to \tau) =< 1, \#(\sigma), \#(\tau) >$. □

The function $\#$ is an injective coding of the types in ω.

4.1.8. DEFINITION. The systems $\lambda\mathsf{E}(\mathcal{N})$ and $\lambda\mathsf{E}\eta(\mathcal{N})$ are defined as follows.

(i) The constants of $\lambda\mathsf{E}(\mathcal{N})$, their types and their arities are given by the following table.

constant	type	arity
$\mathbf{E}_\sigma$ $(\sigma \in \mathbb{T})$	$0 \to \sigma$	1
$\mathbf{c}_m$ $(m \in \omega)$	0	0
$\mathbf{S}^+$	$0 \to 0$	1
$\mathbf{\Delta}$	$t(4)$	2

By this table we already obtain the set $\mathcal{T}(\lambda\mathsf{E}(\mathcal{N}))$ of all $\lambda\mathsf{E}(\mathcal{N})$-terms.

(ii) Define the coding $\# : \mathcal{T}(\lambda\mathsf{E}(\mathcal{N})) \to \omega$ by induction on the structure of the $\lambda\mathsf{E}(\mathcal{N})$-terms as follows.

(1) $\#(v_n^\sigma) =< 0, \#(\sigma), n >$.

(2) $\#(\mathbf{E}_\sigma) =< 1, \#(\sigma) >$.

(3) $\#(\mathbf{c}_m) =< 2, m >$.

(4) $\#(\mathbf{S}^+) =< 3 >$.

(5) $\#(\mathbf{\Delta}) =< 4 >$.

(6) $\#(MN) =< 5, \#(M), \#(N) >$.

(7) $\#(\lambda x.M) =< 6, \#(x), \#(M) >$.

(iii) Let ε be the reductor on $\mathcal{T}(\lambda\mathsf{E}(\mathcal{N}))$ consisting of the following reduction rules:

$$\mathbf{E}_\sigma \mathbf{c}_{\# M} \to M$$

for all $M \in \mathcal{T}_\sigma(\lambda\mathsf{E}(\mathcal{N}))^0$.

(iv) Let α be the reductor on $\mathcal{T}(\lambda\mathsf{E}(\mathcal{N}))$ consisting of the following reduction rules:

(1) $\mathbf{S}^+\mathbf{c}_m \to_\alpha \mathbf{c}_{m+1}$.

(2) $\mathbf{\Delta}\mathbf{c}_m\mathbf{c}_m \to_\alpha \mathbf{true}_0$.

(3) $\mathbf{\Delta}\mathbf{c}_m\mathbf{c}_n \to_\alpha \mathbf{false}_0$ if $m \neq n$.

(v) $\lambda\mathsf{E}(\mathcal{N}) = (\mathcal{T}(\lambda\mathsf{E}(\mathcal{N})), \alpha\beta\varepsilon)$.

(vi) $\lambda\mathsf{E}\eta(\mathcal{N}) = (\mathcal{T}(\lambda\mathsf{E}(\mathcal{N})), \alpha\beta\varepsilon\eta)$. □

4.1.9. THEOREM. (i) $\lambda\mathsf{E}(\mathcal{N}) \models \mathbf{CR}$.

(ii) $\lambda\mathsf{E}\eta(\mathcal{N}) \models \mathbf{CR}$.

PROOF. By the First Church-Rosser Theorem and the fact that $\alpha\varepsilon$-reduction is deterministic, closed and $\alpha\beta\varepsilon\eta$-normal. □

From this theorem it follows that the sequence $\mathbf{c} = (\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2, \ldots)$ is a numeral system in $\lambda\mathsf{E}(\mathcal{N})$ and in $\lambda\mathsf{E}\eta(\mathcal{N})$.

4.1.10. THEOREM. (i) *The systems $\lambda\mathsf{E}(\mathcal{N})$ and $\lambda\mathsf{E}\eta(\mathcal{N})$ both have enumerators w.r.t.* $\mathbf{c}$.

(ii) *The systems $\lambda\mathsf{E}(\mathcal{N})$ and $\lambda\mathsf{E}\eta(\mathcal{N})$ both have fixed point combinators.*

PROOF. (i) Simple.

(ii) By 4.1.2. □

4.1.11. THEOREM. (i) $\lambda\mathsf{Y}(\mathcal{N}) \trianglelefteq \lambda\mathsf{E}(\mathcal{N})$.

(ii) *Every recursive function on ω is $\lambda\mathsf{E}(\mathcal{N})$-computable w.r.t.* $\mathbf{c}$.

PROOF. For every $\sigma \in \mathbb{T}$ let Y_σ be a fixed point combinator in $\lambda\mathsf{E}(\mathcal{N})$ of type $(\sigma \to \sigma) \to \sigma$. Let $\mathbb{C}$ be a constant family with $\mathcal{T}(\mathbb{C}) = \mathcal{T}(\lambda\mathsf{Y}(\mathcal{N}))$ and define $\xi : |\mathbb{C}| \to \mathcal{T}(\lambda\mathsf{E}(\mathcal{N}))$ by

(i) $\xi(\mathbf{Y}_\sigma) \equiv \mathsf{Y}_\sigma$.

(ii) $\xi(\mathbf{c}_m) \equiv \mathbf{c}_m$.

(iii) $\xi(\mathbf{S}^+) \equiv \mathbf{S}^+$.

(iv) $\xi(\mathbf{\Delta}) \equiv \mathbf{\Delta}$.

Then Ξ_ξ is a translation from $\lambda\mathsf{Y}(\mathcal{N})$ into $\lambda\mathsf{E}(\mathcal{N})$, and so $\lambda\mathsf{Y}(\mathcal{N}) \trianglelefteq \lambda\mathsf{E}(\mathcal{N})$. Also $\Xi_\xi(\mathbf{c}_m) \equiv \mathbf{c}_m$ for all $m \in \omega$, so every function on ω that is $\lambda\mathsf{Y}(\mathcal{N})$-computable w.r.t. $\mathbf{c}$ is also $\lambda\mathsf{E}(\mathcal{N})$-computable w.r.t. $\mathbf{c}$. But every recursive function is $\lambda\mathsf{Y}(\mathcal{N})$-computable, so every recursive function is also $\lambda\mathsf{E}(\mathcal{N})$-computable. $\Box$

If $\Xi : \lambda\mathsf{Y}(\mathcal{N}) \to \lambda\mathsf{E}(\mathcal{N})$, then every quasi-$\lambda\mathsf{E}(\mathcal{N})$-algebra $(\mathfrak{M}, [\![\;]\!]^1)$ can be transformed into a quasi-$\lambda\mathsf{Y}(\mathcal{N})$-algebra $(\mathfrak{M}, [\![\;]\!]^2)$ by defining

$$[\![M]\!]^2_\rho = [\![\Xi(M)]\!]^1_\rho$$

for all $M \in \mathcal{T}(\lambda\mathsf{Y}(\mathcal{N}))$. Moreover, if $(\mathfrak{M}, [\![]\!]^1)$ is a $\lambda\mathsf{E}(\mathcal{N})$-algebra, a $\lambda\mathsf{E}(\mathcal{N})$-model or a quasi-$\lambda\mathsf{E}(\mathcal{N})$-model, then $(\mathfrak{M}, [\![]\!]^2)$ is a $\lambda\mathsf{Y}(\mathcal{N})$-algebra, a $\lambda\mathsf{Y}(\mathcal{N})$-model or a quasi-$\lambda\mathsf{Y}(\mathcal{N})$-model.

Later in this chapter we will characterize the quasi-extensional quasi-$\lambda\mathsf{E}(\mathcal{N})$-models using recursion theories on type structures, and we will show that the type structure **MON** can be made into a $\lambda\mathsf{E}(\mathcal{N})$-model by first constructing a recursion theory on **MON**.

The next theorem is a general result on the number of type-0 objects in quasi-agebras for systems that have enumerators.

4.1.12. THEOREM. *Let Σ be a system that has enumerators w.r.t. a numeral system* $\mathbf{a}$. *Let $(\mathfrak{M}, [\![\;]\!])$ be a quasi-Σ-algebra. Then $\mathfrak{M}_0$ is either infinite or a singleton.*

PROOF. Let $\mathsf{E}_{0\to0}$ be the enumerator for type $0 \to 0$ in Σ. For every $m \in \omega$ define $m^* = [\![\mathbf{a}_m]\!]$. Also define $A = \{m^* | \; m \in \omega \}$.

The idea of this proof is the following. Define $\mathcal{E} : A \to \mathfrak{M}_{0\to0}$ by $\mathcal{E}(a) = [\![\mathsf{E}_{0\to0}]\!]a$. Let $\mathfrak{D} = \mathfrak{D}_{0\to0}(\mathfrak{M}, [\![\;]\!])$ be the set of all definable type-$(0 \to 0)$ objects in $(\mathfrak{M}, [\![\;]\!])$. Then $\mathcal{E} : A \twoheadrightarrow \mathfrak{D}$ so $\mathbf{card}(\mathfrak{D}) \leq \mathbf{card}(A)$. If we suppose that $\mathfrak{M}_0$ is finite but not a singleton, then it follows that A is finite, say $\mathbf{card}(A) = k$ and $A = \{a_1, \ldots, a_k\}$, and that $\lambda x.a_1, \ldots, \lambda x.a_k$ and $\lambda x.x$ are $k+1$ different elements of $\mathfrak{D}$, so $\mathbf{card}(\mathfrak{D}) \geq k+1 > \mathbf{card}(A)$. Contradiction.

The precise proof is as follows. Suppose $\mathfrak{M}_0$ is finite. We are going to show that $\mathfrak{M}_0$ is a singleton. Because $A \subseteq \mathfrak{M}_0$, also A is finite, so there exist $k \in \omega, k \geq 1$ and $m_1, \ldots, m_k \in \omega$ such that
(i) $A = \{m_1^*, \ldots, m_k^*\}$
(ii) $m_p^* = m_q^* \Rightarrow p = q$.
Then $\mathbf{card}(A) = k$. For every $p \in \{1, \ldots, k\}$ define $M_p \equiv \lambda x.\mathbf{a}_{m_p}$ (with $x \in Var_0$). Then $M_p \in \mathcal{T}_{0\to 0}(\Sigma)^0$ $(p = 1, \ldots, k)$. So there exist $n_1, \ldots, n_k \in \omega$ such that

$$\Sigma \vdash \mathsf{E}_{0\to 0}\mathbf{a}_{n_p} = M_p \quad (p = 1, \ldots, k).$$

Because $(\mathfrak{M}, [\![\;]\!])$ is a quasi-Σ-algebra, this gives

$$\begin{aligned}
[\![M_p]\!] &\approx [\![\mathsf{E}_{0\to 0}\mathbf{a}_{n_p}]\!] \\
&= [\![\mathsf{E}_{0\to 0}]\!][\![\mathbf{a}_{n_p}]\!] \\
&= [\![\mathsf{E}_{0\to 0}]\!]n_p^*
\end{aligned}$$

for every $p \in \{1, \ldots, k\}$. Now let $p, q \in \{1, \ldots, k\}$ be such that $n_p^* = n_q^*$. Then

$$[\![M_p]\!] \approx [\![\mathsf{E}_{0\to 0}]\!]n_p^* = [\![\mathsf{E}_{0\to 0}]\!]n_q^* \approx [\![M_q]\!]$$

so $[\![M_p]\!] \approx [\![M_q]\!]$. Let $a \in \mathfrak{M}_0$. Then

$$\begin{aligned}
m_p^* &= [\![\mathbf{a}_{m_p}]\!] \\
&= [\![M_p]\!]a \\
&= [\![M_q]\!]a \\
&= [\![\mathbf{a}_{m_q}]\!] \\
&= m_q^*.
\end{aligned}$$

But then, by (ii) above $p = q$. So $\mathbf{card}(\{n_1^*, \ldots, n_k^*\}) = k$. Also $\{n_1^*, \ldots, n_k^*\} \subseteq A$ and $\mathbf{card}(A) = k$, so $A = \{n_1^*, \ldots, n_k^*\}$. Because $\mathbf{I}_0 \equiv \lambda x.x \in \mathcal{T}_{0\to 0}(\Sigma)^0$ there exists $n \in \omega$ such that $\Sigma \vdash \mathsf{E}_{0\to 0}\mathbf{a}_n = \mathbf{I}_0$. Then $n^* \in A$, so $n^* = n_p^*$ for some $p \in \{1, \ldots, k\}$. Then

$$\begin{aligned}
[\![\mathbf{I}_0]\!] &\approx [\![\mathsf{E}_{0\to 0}\mathbf{a}_n]\!] \\
&= [\![\mathsf{E}_{0\to 0}]\!][\![\mathbf{a}_n]\!] \\
&= [\![\mathsf{E}_{0\to 0}]\!]n^* \\
&= [\![\mathsf{E}_{0\to o}]\!]n_p^* \\
&\approx [\![M_p]\!]
\end{aligned}$$

so $[\![\mathbf{I}_0]\!] \approx [\![M_p]\!]$. So for all $b \in \mathfrak{M}_0$

$$b = [\![\mathbf{I}_0]\!]b = [\![M_p]\!]b = [\![\mathbf{a}_{m_p}]\!] = m_p^*$$

so $\mathfrak{M}_0 \subseteq \{m_p^*\}$, and so $\mathfrak{M}_0 = \{m_p^*\}$. $\square$

To end this section we will construct a translation of $\lambda\mathsf{E}(\mathcal{N})$ into the type-free λ-calculus λ.

4.1.13. Definition. (i) The set of all type-free variables:

$$Var = \{v_0, v_1, v_2, \ldots\}.$$

(ii) The set Λ of all type-free λ-terms is defined inductively as follows.

(1) $Var \subseteq \Lambda$.

(2) $M, N \in \Lambda \Rightarrow (MN) \in \Lambda$.

(3) $x \in Var,\ M \in \Lambda \Rightarrow (\lambda x.M) \in \Lambda$.

(iii) Let β be the reductor on Λ consisting of the following reduction rules:

$$(\lambda x.M)N \to M[x := N].$$

This gives a reduction relation $\twoheadrightarrow_\beta$ on Λ called β-reduction and a conversion relation $=_\beta$ on Λ called β-conversion or β-equality.

(iv) The (pure) type-free λ-calculus is the system $\lambda = (\Lambda, \beta)$.

(v) The set of all closed λ-terms in Λ is denoted by Λ^0. $\square$

In λ we have pairing, booleans and numerals.

4.1.14. Definition. (i) For all $M, N \in \Lambda$ define

$$[M, N] \equiv \lambda x.xMN.$$

(ii) $\mathbf{true} \equiv \lambda xy.x$.

(iii) $\mathbf{false} \equiv \lambda xy.y$.

(iv) $\mathbf{I} \equiv \lambda x.x$.

(v) For every $m \in \omega$ the numeral $\ulcorner m \urcorner \in \Lambda^0$ is defined by induction on m as follows.

(1) $\ulcorner 0 \urcorner \equiv \mathbf{I}$.

(2) $\ulcorner m+1 \urcorner \equiv [\mathbf{false}, \ulcorner m \urcorner]$. $\square$

The following three theorems are borrowed from the type-free λ-calculus. Their proofs can be found in [Barendregt 1984].

4.1.15. Theorem. *Every recursive function on ω is λ-computable w.r.t. the numerals $\ulcorner 0 \urcorner, \ulcorner 1 \urcorner, \ulcorner 2 \urcorner, \ldots$: for every recursive function $f : \omega^k \to \omega$ there exists $F \in \Lambda^0$ such that for all $m_1, \ldots, m_k \in \omega$*

$$F \ulcorner m_1 \urcorner \ldots \ulcorner m_k \urcorner =_\beta \ulcorner f(m_1, \ldots, m_k) \urcorner. \ \square$$

4.1.16. THEOREM (Enumeration Theorem).
There exists $\mathbf{E} \in \Lambda^0$ *such that for all* $M \in \Lambda^0$

$$\mathbf{E}\ulcorner\#M\urcorner =_\beta M$$

where $\# : \Lambda \rightarrowtail \omega$ *is a coding of type-free* λ*-terms.* □

4.1.17. THEOREM (Second Fixed Point Theorem).
For every $F \in \Lambda^0$ *there exists* $M \in \Lambda^0$ *such that*

$$F\ulcorner\#M\urcorner =_\beta M.$$

4.1.18. DEFINITION. (i) A transformation from $\mathcal{T}(\mathbb{C})$ into Λ is a mapping $|\ | : \mathcal{T}(\mathbb{C}) \to \Lambda$ such that

(1) $x \in Var_{\mathbb{T}} \Rightarrow |x| \in Var$.

(2) $|x| \equiv |y| \Rightarrow x \equiv y$.

(3) $\mathbf{C} \in |\mathbb{C}| \Rightarrow |\mathbf{C}| \in \Lambda^0$.

(4) $|MN| \equiv |M||N|$.

(5) $|\lambda x.M| \equiv \lambda|x|.|M|$.

(ii) A translation from Σ into λ is a transformation $|\ | : \mathcal{T}(\Sigma) \to \Lambda$ such that for all $M, N \in \mathcal{T}(\Sigma)$

$$\Sigma \vdash M = N \Rightarrow |M| =_\beta |N|. \ \Box$$

4.1.19. LEMMA. (i) *Let* $|\ | : \mathcal{T}(\Sigma) \to \Lambda$ *be a transformation such that for every* $(M = N) \in Ax(\Sigma)$ *we have* $|M| =_\beta |N|$. *Then* $|\ |$ *is a translation from* Σ *into* λ.

(ii) *Let* Σ *be a* $\beta\delta$*-system and let* $|\ | : \mathcal{T}(\Sigma) \to \Lambda$ *be a transformation such that for every* $(M = N) \in Ax_\delta(\Sigma)$ *we have* $|M| =_\beta |N|$. *Then* $|\ |$ *is a translation from* Σ *into* λ.

PROOF. (i) Simple.

(ii) Using induction on the structure of M it can be shown that

$$|M[x := N]| \equiv |M|[x := |N|].$$

From this it follows that

$$|(\lambda x.M)N| =_\beta |M[x := N]|.$$

So for every Σ-axiom $(M = N)$ we have $|M| =_\beta |N|$, and therefore it follows by (i) that $|\ |$ is a translation from Σ into λ. □

4.1.20. THEOREM. *There exists a translation from* $\lambda\mathsf{E}(\mathcal{N})$ *into* λ.

PROOF. Let $\mathbf{Sc} \in \Lambda^0$ be such that for all $m \in \omega$

$$\mathbf{Sc}\ulcorner m \urcorner =_\beta \ulcorner m+1 \urcorner.$$

Let $\mathbf{Eq} \in \Lambda^0$ be such that for all $m, n \in \omega$

- $\mathbf{Eq}\ulcorner m \urcorner\ulcorner n \urcorner =_\beta \mathbf{true}$ if $m = n$
- $\mathbf{Eq}\ulcorner m \urcorner\ulcorner n \urcorner =_\beta \mathbf{false}$ if $m \neq n$.

Now define, for every $F \in \Lambda^0$, a mapping

$$| \ |_F : \mathcal{T}(\lambda\mathsf{E}(\mathcal{N})) \to \Lambda$$
$$M \mapsto |M|_F$$

by induction on the structure of M as follows.
(i) $|\mathbf{E}_\sigma|_F \equiv \mathbf{E} \circ F (\equiv \lambda x.\mathbf{E}(Fx))$.
(ii) $|\mathbf{c}_m|_F \equiv \ulcorner m \urcorner$.
(iii) $|\mathbf{S}^+|_F \equiv \mathbf{Sc}$.
(iv) $|\mathbf{\Delta}|_F \equiv \mathbf{Eq}$.
(v) $|v_n^\sigma|_F \equiv v_{<\#(\sigma),n>}$.
(vi) $|MN|_F \equiv |M|_F|N|_F$.
(vii) $|\lambda x.M|_F \equiv \lambda|x|_F.|M|_F$.
By the effectiveness of the map

$$(F, M) \mapsto |M|_F$$

there exists a recursive $g : \omega^2 \to \omega$ such that for all $F \in \Lambda^0$ and $M \in \mathcal{T}(\lambda\mathsf{E}(\mathcal{N}))$

$$g(\#(F), \#(M)) = \#(|M|_F).$$

Then there exists $G \in \Lambda^0$ such that G λ-computes g w.r.t. $\ulcorner 0 \urcorner, \ulcorner 1 \urcorner, \ulcorner 2 \urcorner, \ldots$ By the Second Fixed Point Theorem there exists $F \in \Lambda^0$ with $G\ulcorner \#(F) \urcorner =_\beta F$. Then for all $M \in \mathcal{T}(\lambda\mathsf{E}(\mathcal{N}))$

$$F\ulcorner \# M \urcorner =_\beta G\ulcorner \# F \urcorner\ulcorner \# M \urcorner =_\beta \ulcorner g(\#F, \#M) \urcorner =_\beta \ulcorner \#(|M|_F) \urcorner.$$

The map $| \ |_F$ is a transformation from $\mathcal{T}(\lambda\mathsf{E}(\mathcal{N}))$ into Λ. For all $\sigma \in \mathbb{T}$ and $M \in \mathcal{T}_\sigma(\lambda\mathsf{E}(\mathcal{N}))^0$ we have that in λ

$$\begin{aligned} |\mathbf{E}_\sigma \mathbf{c}_m|_F &= |\mathbf{E}_\sigma|_F |\mathbf{c}_m|_F \\ &= (\mathbf{E} \circ F)\ulcorner \# M \urcorner \\ &= \mathbf{E}(F\ulcorner \# M \urcorner) \\ &= \mathbf{E}\ulcorner \#(|M|_F) \urcorner \\ &= |M|_F \end{aligned}$$

because $|M|_F \in \Lambda^0$. So for every ε-axiom $(M = N)$ of $\lambda\mathsf{E}(\mathcal{N})$ we have $|M|_F =_\beta |N|_F$. It is easy to show that for every α-axiom $(M = N)$ of $\lambda\mathsf{E}(\mathcal{N})$ we also have $|M|_F =_\beta |N|_F$. Therefore $| \ |_F$ is a translation from $\lambda\mathsf{E}(\mathcal{N})$ into λ. $\square$

4.2. Operators on $\mathfrak{M}$-representable functionals.

In the next section we are going to define the notion of recursion theory on a type structure. A recursion theory on a type structure $\mathfrak{M}$ will be a family of $\mathfrak{M}$-representable functionals on $\mathfrak{M}$. In this section we will take a look at a few operators on $\mathfrak{M}$-representable functionals, and at sets of $\mathfrak{M}$-representable functionals that are closed under certain operators.

In the following $\mathfrak{M}$ is a λ-structure and $[\![\,]\!]$ is a quasi-$\lambda^{\rightarrow}$-interpretation in $\mathfrak{M}$. Recall that $\mathbb{F}_\sigma(\mathfrak{M})$ denotes the set of all $\mathfrak{M}$-representable functionals $(\mathfrak{M})_\sigma \to \mathfrak{M}_0$, and that $\mathbb{F}(\mathfrak{M})$ denotes the set of all $\mathfrak{M}$-representable functionals.

4.2.1. DEFINITION. An operator on $\mathbb{F}(\mathfrak{M})$ is a map

$$\Phi : \mathbb{F}_{\sigma_1}(\mathfrak{M}) \times \ldots \times \mathbb{F}_{\sigma_k}(\mathfrak{M}) \to \mathbb{F}_\tau(\mathfrak{M})$$

with $\sigma_1, \ldots, \sigma_k, \tau \in \mathbb{T}$. The number k is called the arity of Φ and the type $\sigma_1 \to \ldots \to \sigma_k \to \tau$ is called the type of Φ. $\Box$

Our first example of an operator is composition.

4.2.2. DEFINITION. Let $\sigma \in \mathbb{T}$, $l \in \omega$. Let $\varphi : (\mathfrak{M})_\sigma \times (\mathfrak{M}_0)^l \to \mathfrak{M}_0$ be $\mathfrak{M}$-representable, and let $\psi_1, \ldots, \psi_l : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ also be $\mathfrak{M}$-representable. The composition of φ and $\psi_1, \ldots, \psi_l$ is the functional $\chi : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ defined by

$$\chi(\vec{a}) = \varphi(\vec{a}, \psi_1(\vec{a}), \ldots, \psi_l(\vec{a})).$$

Notation: $\chi = \mathsf{Comp}_{\sigma,l}[\varphi, \psi_1, \ldots, \psi_l]$. $\Box$

4.2.3. LEMMA. *For all $\sigma \in \mathbb{T}$ and $l \in \omega$*

$$\mathsf{Comp}_{\sigma,l} : \mathbb{F}_{\sigma * t(l)}(\mathfrak{M}) \times \mathbb{F}_\sigma(\mathfrak{M})^l \to \mathbb{F}_\sigma(\mathfrak{M}). \ \Box$$

It can easily be shown that the composition of $\mathfrak{M}$-representable functionals is again $\mathfrak{M}$-representable, but we will not prove this now, because in a moment we will prove a more general result from which this lemma follows immediately.

The next operator is the one that 'trivially' manipulates arguments.

4.2.4. DEFINITION. Let $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0 \in \mathbb{T}$. Let $\vec{s} = (s_1, \ldots, s_l)$ be such that $1 \leq s_1, \ldots s_l \leq k$.

(i) $\sigma(\vec{s}) = \sigma_{s_1} \to \sigma_{s_2} \to \ldots \to \sigma_{s_l} \to 0$.

(ii) For every $\vec{a} = (a_1, \ldots, a_k) \in (\mathfrak{M})_\sigma$ define

$$(\vec{a})_{\vec{s}} = (a_{s_1}, a_{s_2}, \ldots, a_{s_l}) \in (\mathfrak{M})_{\sigma(\vec{s})}.$$

(iii) Let $\varphi : (\mathfrak{M})_{\sigma(\vec{s})} \to \mathfrak{M}_0$. The manipulation of φ with parameters σ and $\vec{s}$ is the functional $\psi : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ defined by

$$\psi(\vec{a}) = \varphi((\vec{a})_{\vec{s}}).$$

Notation: $\psi = \mathsf{Man}_{\sigma,\vec{s}}[\varphi]$. □

4.2.5. LEMMA. *Let $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0$ and let $\vec{s} = (s_1, \ldots, s_l)$ be such that $1 \le s_1, \ldots, s_l \le k$. Then*

$$\mathsf{Man}_{\sigma,\vec{s}} : \mathbb{F}_{\sigma(\vec{s})}(\mathfrak{M}) \to \mathbb{F}_\sigma(\mathfrak{M}).$$

PROOF. We have to show that the manipulation of an $\mathfrak{M}$-representable functional is again $\mathfrak{M}$-representable. Let $\varphi \in \mathbb{F}_{\sigma(\vec{s})}(\mathfrak{M})$ and let $\psi = \mathsf{Man}_{\sigma,\vec{s}}[\varphi]$. Let $f \in \mathfrak{M}_{\sigma(\vec{s})}$ $\mathfrak{M}$-represent φ. Then for all $\vec{a} \in (\mathfrak{M})_\sigma$

$$\begin{array}{rcl} \psi(\vec{a}) & = & \varphi((\vec{a})_{\vec{s}}) \\ & = & f a_{s_1} a_{s_2} \ldots a_{s_l} \\ & = & [\![y x_{s_1} x_{s_2} \ldots x_{s_l}]\!]_{\rho(\vec{x}:=\vec{a})} \\ & = & [\![\lambda \vec{x}. y x_{s_1} x_{s_2} \ldots x_{s_l}]\!]_\rho \vec{a} \end{array}$$

where $\rho \in Val(\mathfrak{M})$ is such that $\rho(y) = f$. So ψ is $\mathfrak{M}$-representable. □

4.2.6. LEMMA. *Let $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0$. Let $i, j_1, \ldots j_l \in \{1, \ldots, k\}$ be such that*

$$\sigma_i = \sigma_{j_1} \to \ldots \to \sigma_{j_l} \to 0.$$

Define $\varphi : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ by

$$\varphi(\vec{a}) = a_i a_{j_1} a_{j_2} \ldots a_{j_l}.$$

Then $\varphi = \mathsf{Man}_{\sigma,\vec{s}}[\mathsf{Ap}_{\sigma_i}]$ and φ is $\mathfrak{M}$-representable.

PROOF. For all $\vec{a} \in (\mathfrak{M})_\sigma$

$$\begin{array}{rcl} \varphi(\vec{a}) & = & a_i a_{j_1} \ldots a_{j_l} \\ & = & \mathsf{Ap}_{\sigma_i}(a_i, a_{j_1}, \ldots, a_{j_l}) \\ & = & \mathsf{Ap}_{\sigma_i}((\vec{a})_{(i,j_1,\ldots,j_l)}) \\ & = & \mathsf{Man}_{\sigma,(i,j_1,\ldots,j_l)}[\mathsf{Ap}_{\sigma_i}](\vec{a}). \ \square \end{array}$$

The next operator is a generalization of composition, called substitution. Closure under substitution is a very important property of the recursion theories that will be introduced in the next section. In order to understand that the substitution operator, which will be defined below, is well-defined, we need extensionality of functionals.

4.2.7. DEFINITION. Let $\sigma \in \mathbb{T}$ and $\varphi : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$. Then φ is extensional if for all $\vec{a}, \vec{b} \in (\mathfrak{M})_\sigma$

$$\vec{a} \approx \vec{b} \Rightarrow \varphi(\vec{a}) = \varphi(\vec{b}).\ \Box$$

Because $\mathfrak{M}$ is quasi-extensional, every $\mathfrak{M}$-representable functional on $\mathfrak{M}$ is extensional.

Before defining substitution we will give an informal explanation of this operator. Let $\sigma, \tau \in \mathbb{T}$, $\tau = \tau_1 \to \ldots \to \tau_l \to 0$. Let $\varphi : (\mathfrak{M})_{\sigma * \tau} \to \mathfrak{M}_0$ be $\mathfrak{M}$-representable and for every $j \in \{1, \ldots l\}$ let $\psi_j : (\mathfrak{M})_{\tau_j} \to \mathfrak{M}_0$ be $\mathfrak{M}$-representable. For a moment fix $j \in \{1, \ldots, l\}$. Let $\vec{a} \in (\mathfrak{M})_\sigma$. Then the functional $\vec{c} \mapsto \psi_j(\vec{a}, \vec{c})$ is $\mathfrak{M}$-representable. The expression

$$\lambda\vec{c}.\psi_j(\vec{a}, \vec{c})$$

denotes an arbitrary type-τ_j object that $\mathfrak{M}$-represents $\vec{c} \mapsto \psi_j(\vec{a}, \vec{c})$. Then for every $b \in \mathfrak{M}_{\tau_j}$ that $\mathfrak{M}$-represents $\vec{c} \mapsto \psi_j(\vec{a}, \vec{c})$ we have

$$b \approx \lambda\vec{c}.\psi_j(\vec{a}, \vec{c}).$$

The substitution of φ and $\psi_1, \ldots \psi_l$ is the functional $\chi : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ defined by

$$\chi(\vec{a}) = \varphi(\vec{a}, \lambda\vec{c}_1.\psi_1(\vec{a}, \vec{c}_1), \ldots, \lambda\vec{c}_l.\psi_l(\vec{a}, \vec{c}_l)).$$

Because φ is extensional, the value of

$$\varphi(\vec{a}, \lambda\vec{c}_1.\psi_1(\vec{a}, \vec{c}_1), \ldots, \lambda\vec{c}_l.\psi_l(\vec{a}, \vec{c}_l))$$

is independent of the particular choices for $\lambda\vec{c}_j.\psi_j(\vec{a}, \vec{c}_j)$, and therefore χ is well-defined.

4.2.8. DEFINITION. Let $\sigma, \tau \in \mathbb{T}$, $\tau = \tau_1 \to \ldots \to \tau_l \to 0$. Let $\varphi \in \mathbb{F}_{\sigma * \tau}(\mathfrak{M})$ and for every $j \in \{1, \ldots, l\}$ let $\psi_j \in \mathbb{F}_{\sigma * \tau_j}(\mathfrak{M})$. The substitution of φ and $\psi_1, \ldots, \psi_l$ is the functional $\chi : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ defined as follows. For all $\vec{a} \in (\mathfrak{M})_\sigma$ and $m \in \mathfrak{M}_0$

$$\chi(\vec{a}) = m$$

if and only if there exists $\vec{b} = (b_1, \ldots, b_l) \in (\mathfrak{M})_\tau$ such that

(i) $\varphi(\vec{a}, \vec{b}) = m$

(ii) $\forall j \in \{1, \ldots, l\} \forall \vec{c} \in (\mathfrak{M})_{\tau_j} [\, \psi_j(\vec{a}, \vec{c}) = b_j\vec{c}\,]$.

Notation: $\chi = \mathsf{Sub}_{\sigma,\tau}[\varphi, \psi_1, \ldots, \psi_l]$. $\Box$

4.2.9. LEMMA. *Let $\sigma, \tau \in \mathbb{T}$, $\tau = \tau_1 \to \ldots \to \tau_l \to 0$. Let $\varphi \in \mathbb{F}_{\sigma * \tau}(\mathfrak{M})$ and $\psi_j \in \mathbb{F}_{\sigma * \tau_j}(\mathfrak{M})$ $(j = 1, \ldots, l)$. Let $\chi = \mathsf{Sub}_{\sigma,\tau}[\varphi, \psi_1, \ldots, \psi_l]$. Let $\vec{a} \in (\mathfrak{M})_\sigma$. Let $\vec{b} \in (\mathfrak{M})_\tau$ be such that $\psi_j(\vec{a}, \vec{c}) = b_j\vec{c}$ for every $j \in \{1, \ldots, l\}$ and all $\vec{c} \in (\mathfrak{M})_{\tau_j}$. Then $\chi(\vec{a}) = \varphi(\vec{a}, \vec{b})$.*

PROOF. Simple. $\Box$

4.2.10. LEMMA. *For all* $\sigma, \tau \in \mathbb{T}$ *with* $\tau = \tau_1 \to \ldots \to \tau_l \to 0$

$$\mathsf{Sub}_{\sigma,\tau} : \mathbb{F}_{\sigma * \tau}(\mathfrak{M}) \times \mathbb{F}_{\sigma * \tau_1}(\mathfrak{M}) \times \ldots \times \mathbb{F}_{\sigma * \tau_l}(\mathfrak{M}) \to \mathbb{F}_{\sigma}(\mathfrak{M}).$$

PROOF. We have to show that the substitution of $\mathfrak{M}$-representable functionals is again $\mathfrak{M}$-representable. Let $\sigma, \tau \in \mathbb{T}$, $\tau = \tau_1 \to \ldots \to \tau_l \to 0$, $\varphi \in \mathbb{F}_{\sigma*\tau}(\mathfrak{M})$, $\psi_j \in \mathbb{F}_{\sigma*\tau_j}(\mathfrak{M})$ $(j = 1, \ldots, l)$ and $\chi = \mathsf{Sub}_{\sigma,\tau}[\varphi, \psi_1, \ldots, \psi_l]$. Let $f \in \mathfrak{M}_{\sigma*\tau}$ $\mathfrak{M}$-represent φ and for every $j \in \{1, \ldots, l\}$ let $g_j \in \mathfrak{M}_{\sigma*\tau_j}$ $\mathfrak{M}$-represent ψ_j. Let $\vec{a} \in (\mathfrak{M})_\sigma$. Define $b_j = g_j\vec{a}$ $(j = 1, \ldots, l)$ and define $\vec{b} = (b_1, \ldots, l)$. Then $\vec{b} \in (\mathfrak{M})_\tau$ and $b_j\vec{c} = \psi_j(\vec{a}, \vec{c})$ for all $j \in \{1, \ldots, l\}$ and $\vec{c} \in (\mathfrak{M})_{\tau_j}$, and so, by the previous lemma

$$\begin{aligned}
\chi(\vec{a}) &= \varphi(\vec{a}, \vec{b}) \\
&= f\vec{a}\vec{b} \\
&= f\vec{a}(g_1\vec{a}) \ldots (g_l\vec{a}) \\
&= [\![y\vec{x}(z_1\vec{x}) \ldots (z_l\vec{x})]\!]_{\rho(\vec{x}:=\vec{a})} \\
&= [\![\lambda\vec{x}, y\vec{x}(z_1\vec{x}) \ldots (z_l\vec{x})]\!]_{\rho}\vec{a}
\end{aligned}$$

where $\rho \in Val(\mathfrak{M})$ is such that $\rho(y) = f$ and $\rho(z_j) = g_j$. Therefore χ is $\mathfrak{M}$-representable. $\Box$

Note that for all $\sigma \in \mathbb{T}$ and $l \in \omega$ we have $\mathsf{Comp}_{\sigma,l} = \mathsf{Sub}_{\sigma,t(l)}$, so by the previous lemma it follows that $\mathbb{F}(\mathfrak{M})$ is closed under composition.

Next we will take a look at sets of $\mathfrak{M}$-representable functionals.

4.2.11. DEFINITION. Let $\mathcal{F} \subseteq \mathbb{F}(\mathfrak{M})$. Then $\mathcal{F}$ is called elementary if $\mathcal{F}$ contains Ap_σ for every $\sigma \in \mathbb{T}$ and is closed under manipulation and substitution. $\Box$

4.2.12. DEFINITION. Let $\mathcal{F} \subseteq \mathbb{F}(\mathfrak{M})$.

(i) For every $\sigma \in \mathbb{T}$ the set of all $\mathcal{F}$-objects of type σ is defined by

$$\mathbb{O}_\sigma(\mathcal{F}) = \{a \in \mathfrak{M}_\sigma \mid fnl(a) \in \mathcal{F}\}.$$

(ii) The set of all $\mathcal{F}$-objects is defined by

$$\mathbb{O}(\mathcal{F}) = \bigcup_{\sigma \in \mathbb{T}} \mathbb{O}_\sigma(\mathcal{F}). \ \Box$$

In the following we will prove that $\mathbb{O}(\mathcal{F})$ is a λ-set for elementary $\mathcal{F}$.

Let $(\mathbf{k}_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}$ be a quasi-k-family in $\mathfrak{M}$ and let $(\mathbf{s}_{\sigma,\tau,\upsilon})_{\sigma,\tau,\upsilon\in\mathbb{T}}$ be a quasi-s-family in $\mathfrak{M}$.

The next lemma is also going to be useful in section 4.4.

4.2.13. LEMMA. *Let* $\sigma, \tau, \upsilon \in \mathbb{T}$, $k = fac(\sigma)$, $l = fac(\tau)$, $m = fac(\upsilon)$.

(i) *Let* $a \in \mathfrak{M}_{\sigma\to\tau}$, $b \in \mathfrak{M}_\sigma$. *Define*

$$\chi_1 = \mathsf{Man}_{\tau*(\sigma\to 0),(l+1,1,2,3,\ldots,l)}[fnl(a)]$$

$$\chi_2 = \mathsf{Man}_{\tau*\sigma,(l+1,l+2,\ldots,l+k)}[fnl(b)]$$

$$\chi = \mathsf{Sub}_{\tau,\sigma\to 0}[\chi_1, \chi_2].$$

Then $\chi = fnl(ab)$.

(ii) *Define* $\chi = \mathsf{Man}_{\sigma\to\tau\to\sigma,(1,3,4,5,\ldots,k+2)}[\mathsf{Ap}_\sigma]$. *Then* $\chi = fnl(\mathbf{k}_{\sigma,\tau})$.

(iii) *Define* $\sigma_0 = (\sigma\to\tau\to\upsilon)\to(\sigma\to\tau)\to\sigma\to\upsilon$, *and define*

$$\chi_1 = \mathsf{Man}_{\sigma_0*(\tau\to 0),(1,3,m+4,4,5,6,\ldots,m+3)}[\mathsf{Ap}_{\sigma\to\tau\to\upsilon}]$$

$$\chi_2 = \mathsf{Man}_{\sigma_0*\tau,(2,3,m+4,m+5,\ldots,m+l+3)}[\mathsf{Ap}_{\sigma\to\tau}]$$

$$\chi = \mathsf{Sub}_{\sigma_0,\tau\to 0}[\chi_1, \chi_2].$$

Then $\chi = fnl(\mathbf{s}_{\sigma,\tau,\upsilon})$.

PROOF. (i) We have $\chi_1 : (\mathfrak{M})_\tau \times \mathfrak{M}_\sigma \to \mathfrak{M}_0$ and $\chi_2 : (\mathfrak{M})_\tau \times (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ with

$$\chi_1(\vec{c}, d) = ad\vec{c}$$

$$\chi_2(\vec{c}, \vec{e}) = b\vec{e}$$

so for all $\vec{c} \in (\mathfrak{M})_\tau$ we have

$$\chi(\vec{c}) = \chi_1(\vec{c}, b) = ab\vec{c}$$

and so $\chi = fnl(ab)$.

(ii) We have $\chi : \mathfrak{M}_\sigma \times \mathfrak{M}_\tau \times (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ with

$$\chi(a, b, \vec{c}) = a\vec{c} = \mathbf{k}_{\sigma,\tau}ab\vec{c}$$

and so $\chi = fnl(\mathbf{k}_{\sigma,\tau})$.

(iii) We have $\chi_1 : (\mathfrak{M})_{\sigma_0} \times \mathfrak{M}_\tau \to \mathfrak{M}_0$ and $\chi_2 : (\mathfrak{M})_{\sigma_0} \times (\mathfrak{M})_\tau \to \mathfrak{M}_0$ with

$$\chi_1(a, b, c, \vec{d}, f) = acf\vec{d}$$

$$\chi_2(a, b, c, \vec{d}, \vec{e}) = bc\vec{e}$$

so

$$\begin{aligned} \chi(a, b, c, \vec{d}) &= \chi_1(a, b, c, \vec{d}, bc) \\ &= ac(bc)\vec{d} \\ &= \mathbf{s}_{\sigma,\tau,\upsilon}abc\vec{d} \end{aligned}$$

and so $\chi = fnl(\mathbf{s}_{\sigma,\tau,\upsilon})$. □

4.2.14. PROPOSITION. *If* $\mathcal{F} \subseteq \mathbb{F}(\mathfrak{M})$ *is elementary, then* $\mathbb{O}(\mathcal{F})$ *is a* λ*-set.*

PROOF. By the previous lemma. □

4.3. Recursion theories on type structures.

In this section we will axiomatically introduce the notion of recursion theory on a type structure, and we will prove a number of properties of these recursion theories, among which a First Recursion Theorem and a Second Recursion Theorem.

These recursion theories should not be confused with S1-S9 recursion theories on type structures. An S1-S9 recursion theory on a type structure $\mathfrak{M}$ consists of a collection of partial functionals on $\mathfrak{M}$, whereas the recursion theories that we are interested in consist of total $\mathfrak{M}$-representable functionals. In section 4.5 we will however use an S1-S9-like mechanism to construct a recursion theory on the type structure **MON**. For more on S1-S9 recursion see [Kleene 1959], [Gandy and Hyland 1977] or [Kechris and Moschovakis 1977].

We are going to define the notion of recursion theory on so-called ω-structures. These will now be defined.

4.3.1. DEFINITION. An ω-structure is a λ-structure $\mathfrak{M}$ with $\omega \subseteq \mathfrak{M}_0$. □

4.3.2. EXAMPLES. (i) **MON** is an ω-structure.
(ii) The full type structure $\mathfrak{M}(\omega)$ over ω is an ω-structure. □

Let $\mathfrak{M}$ be an ω-structure. Then $\mathfrak{M}$ is a λ-structure, so we have operators $\mathsf{Sub}_{\sigma,\tau}$ and $\mathsf{Man}_{\sigma,\vec{s}}$ on $\mathbb{F}(\mathfrak{M})$ for substitution and manipulation, and we have that $\mathsf{Ap}_\sigma \in \mathbb{F}(\mathfrak{M})$ for all $\sigma \in \mathbb{T}$.

A recursion theory on $\mathfrak{M}$ will be a map $\mathcal{R} : \omega \times \mathbb{T} \to \mathbb{F}(\mathfrak{M})$ with certain properties to be described below, such that

$$\mathcal{R}(e,\sigma) : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$$

for all $e \in \omega$ and $\sigma \in \mathbb{T}$. For all $e \in \omega$ and $\sigma \in \mathbb{T}$ we write

$$\{e\}^\sigma = \mathcal{R}(e,\sigma).$$

So $\{e\}^\sigma : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ is an $\mathfrak{M}$-representable functional for every $e \in \omega$ and every $\sigma \in \mathbb{T}$.

In a moment we will list a number of axioms for $\mathcal{R}$ that constitute the notion of recursion theory.

4.3.3. DEFINITION. (i) Let $\sigma \in \mathbb{T}$. The set of all $\mathcal{R}$-recursive functionals of type σ is defined by

$$\mathbf{REC}_\sigma(\mathcal{R}) = \{\{e\}^\sigma \mid e \in \omega\}.$$

(ii) The set of all $\mathcal{R}$-recursive functionals is defined by

$$\mathbf{REC}(\mathcal{R}) = \bigcup_{\sigma \in \mathbb{T}} \mathbf{REC}_\sigma(\mathcal{R}). \ \square$$

For $\mathcal{R}$ to be a recursion theory we will also assume that there exists a recursive set $\mathbf{Ix} \subseteq \omega$ of indices and a recursive function $\mathbf{type} : \omega \to \omega$ with

$$\forall e \in \mathbf{Ix} \exists \sigma \in \mathbb{T}[\,\mathbf{type}(e) = \#\sigma\,]$$

such that for every $\sigma \in \mathbb{T}$ and every $\varphi \in \mathbf{REC}_\sigma(\mathcal{R})$ there exists $e \in \mathbf{Ix}$ with $\varphi = \{e\}^\sigma$ and $\mathbf{type}(e) = \#\sigma$.

4.3.4. Definition. For every $\sigma \in \mathbb{T}$ define

$$\mathbf{Ix}(\sigma) = \{e \in \mathbf{Ix} |\ \mathbf{type}(e) = \#\sigma\,\}. \ \square$$

Then for every $\sigma \in \mathbb{T}$ and every $\varphi \in \mathbf{REC}_\sigma(\mathcal{R})$ there exists $e \in \mathbf{Ix}(\sigma)$ with $\varphi = \{e\}^\sigma$.

4.3.5. Definition. (i) $\mathcal{R}$ has the Application Property if there exists a recursive function $\mathbf{ap} : \omega \to \omega$ such that for every $\sigma \in \mathbb{T}$

(1) $\mathbf{ap}(\#\sigma) \in \mathbf{Ix}(\sigma)$

(2) $\mathsf{Ap}_\sigma = \{\mathbf{ap}(\#\sigma)\}^{\sigma \to \sigma}$.

(ii) $\mathcal{R}$ has the Substitution Property if there exists a recursive function $\mathbf{sub} : \omega^3 \to \omega$ such that for all $\sigma, \tau \in \mathbb{T}$ with $\tau = \tau_1 \to \ldots \to \tau_l \to 0$, all $e_0 \in \mathbf{Ix}(\sigma * \tau)$ and all $e_j \in \mathbf{Ix}(\sigma * \tau_j)$ $(j = 1, \ldots, l)$

(1) $\mathbf{sub}(\#\sigma, \#\tau, < e_0, e_1, \ldots, e_l >) \in \mathbf{Ix}(\sigma)$

(2) $\{\mathbf{sub}(\#\sigma, \#\tau, < e_0, e_1, \ldots, e_l >)\}^\sigma = \mathsf{Sub}_{\sigma,\tau}[\{e_0\}^{\sigma*\tau}, \{e_1\}^{\sigma*\tau_1}, \ldots, \{e_l\}^{\sigma*\tau_l}]$.

(iii) $\mathcal{R}$ has the Manipulation Property if there exists a recursive function $\mathbf{man} : \omega^3 \to \omega$ such that for all $\sigma \in \mathbb{T}$ with $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0$, all $\vec{s} = (s_1, \ldots, s_l)$ with $1 \leq s_1, \ldots, s_l \leq k$, and all $e \in \mathbf{Ix}(\sigma(\vec{s}))$

(1) $\mathbf{man}(\#\sigma, < \vec{s} >, e) \in \mathbf{Ix}(\sigma)$

(2) $\{\mathbf{man}(\#\sigma, < \vec{s} >, e)\}^\sigma = \mathsf{Man}_{\sigma,\vec{s}}[\{e\}^{\sigma(\vec{s})}]$.

(iv) $\mathcal{R}$ has the Enumeration Property if there exists a recursive function $\mathbf{enum} : \omega \to \omega$ such that for every $\sigma \in \mathbb{T}$

$$\mathbf{enum}(\#\sigma) \in \mathbf{Ix}(0 \to \sigma)$$

and for all $e \in \mathbf{Ix}(\sigma)$ and $\vec{a} \in (\mathfrak{M})_\sigma$

$$\{\mathbf{enum}(\#\sigma)\}^{0 \to \sigma}(e, \vec{a}) = \{e\}^\sigma(\vec{a}). \ \square$$

For every $\sigma \in \mathbb{T}$ and every $m \in \omega$ we have the constant-m functional $\mathsf{C}^\sigma_m : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$. Note that C^σ_m is $\mathfrak{M}$-representable for all $\sigma \in \mathbb{T}$ and $m \in \omega$.

4.3.6. Definition. $\mathcal{R}$ is a recursion theory on $\mathfrak{M}$ if the following axioms (i)-(vii) hold.

(i) C^0_0 is $\mathfrak{M}$-representable.

(ii) There exists an $\mathcal{R}$-recursive function $\mathsf{S}^+ : \mathfrak{M}_0 \to \mathfrak{M}_0$ such that for all $m \in \omega$

$$\mathsf{S}^+(m) = m + 1.$$

(iii) There exists an $\mathcal{R}$-recursive function $\Delta : (\mathfrak{M}_0)^4 \to \mathfrak{M}_0$ such that for all $m, n \in \omega$ and $a, b \in \mathfrak{M}_0$

(1) $\Delta(m, n, a, b) = a$ if $m = n$

(2) $\Delta(m, n, a, b) = b$ if $m \neq n$.

(iv) $\mathcal{R}$ has the Application Property.

(v) $\mathcal{R}$ has the Substitution Property.

(vi) $\mathcal{R}$ has the Manipulation Property.

(vii) $\mathcal{R}$ has the Enumeration Property. □

4.3.7. Definition. The ω-structure $\mathfrak{M}$ is called special if there exists a recursion theory on $\mathfrak{M}$. □

In this section we are going to prove a number of properties of recursion theories. In the next section we will characterize the special ω-structures in terms of $\lambda\mathsf{E}(\mathcal{N})$, and in section 4.5 we will construct a recursion theory on **MON**, thus showing that **MON** is special.

In the remainder of this section $\mathcal{R}$ is assumed to be a recursion theory on $\mathfrak{M}$. Fix recursive functions **ap**, **sub**, **man** and **enum** with the properties described before. Also fix $\mathcal{R}$-recursive S^+ and Δ as described in axioms (ii) and (iii) of the definition of recursion theory.

4.3.8. Theorem. $\mathbf{REC}(\mathcal{R})$ *contains all application functionals and is closed under substitution and manipulation, so* $\mathbf{REC}(\mathcal{R})$ *is elementary.*

Proof. By axioms (iv), (v) and (vi) of the notion of recursion theory. □

4.3.9. Theorem. *For every* $\sigma \in \mathbb{T}$ *define*

$$\mathcal{E}_\sigma = \{\mathbf{enum}(\#\sigma)\}^{0\to\sigma}.$$

Then $\mathcal{E}_\sigma : \mathfrak{M}_0 \times (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ *is* $\mathcal{R}$*-recursive, and for all* $e \in \mathbf{Ix}(\sigma)$ *and* $\vec{a} \in (\mathfrak{M})_\sigma$

$$\mathcal{E}_\sigma(e, \vec{a}) = \{e\}^\sigma(\vec{a}).$$

PROOF. By axiom (vii), the Enumeration Property. □

4.3.10. THEOREM. (i) *There exists a recursive function* $\mathbf{comp} : \omega^3 \to \omega$ *such that for all* $\sigma \in \mathbb{T}$, $l \in \omega$, $e_0 \in \mathbf{Ix}(\sigma * t(l))$ *and* $e_1, \ldots, e_l \in \mathbf{Ix}(\sigma)$

(1) $\mathbf{comp}(\#\sigma, l, < e_0, e_1, \ldots, e_l >) \in \mathbf{Ix}(\sigma)$

(2) $\{\mathbf{comp}(\#\sigma, l, < e_0, e_1, \ldots, e_l >)\}^\sigma = \mathsf{Comp}_{\sigma,l}[\{e_0\}^{\sigma * t(l)}, \{e_1\}^\sigma, \ldots, \{e_l\}^\sigma]$.

(ii) $\mathbf{REC}(\mathcal{R})$ *is closed under composition.*

PROOF. (i) By axiom (v) and the fact that $\mathsf{Comp}_{\sigma,l} = \mathsf{Sub}_{\sigma,t(l)}$.

(ii) By (i). □

4.3.11. THEOREM. (i) *There exists a recursive function* $\mathbf{con} : \omega^2 \to \omega$ *such that for all* $\sigma \in \mathbb{T}$ *and* $m \in \omega$

(1) $\mathbf{con}(\#\sigma, m) \in \mathbf{Ix}(\sigma)$

(2) $\mathsf{C}^\sigma_m = \{\mathbf{con}(\#\sigma, m)\}^\sigma$.

(ii) C^σ_m *is* $\mathcal{R}$*-recursive for all* $\sigma \in \mathbb{T}$ *and* $m \in \omega$.

PROOF. (i) First we define a recursive function $\mathbf{con}_0 : \omega \to \omega$ such that $\mathsf{C}^0_m = \{\mathbf{con}_0(m)\}^0$ for every $m \in \omega$. This can be done by primitive recursion, using the fact that

$$\mathsf{C}^0_{m+1} = \mathsf{S}^+ \circ \mathsf{C}^0_m = \mathsf{Comp}_{0,1}[\mathsf{S}^+, \mathsf{C}^0_m].$$

Then we note that for all $\sigma \in \mathbb{T}$ and $m \in \omega$

$$\begin{aligned} \mathsf{C}^\sigma_m &= \mathsf{Man}_{\sigma,()}[\mathsf{C}^0_m] \\ &= \mathsf{Man}_{\sigma,()}[\{\mathbf{con}_0(m)\}] \\ &= \{\mathbf{man}(\#\sigma, <>, \mathbf{con}_0(m))\}^\sigma. \end{aligned}$$

So take $\mathbf{con}(s, m) = \mathbf{man}(s, <>, \mathbf{con}_0(m))$.

(ii) By (i). □

If $\sigma \in \mathbb{T}$ and $a \in \mathfrak{M}_\sigma$ then a is called $\mathcal{R}$-recursive if the functional $fnl(a)$ is $\mathcal{R}$-recursive. The set of all $\mathcal{R}$-recursive type-σ objects is $\mathbf{Rec}_\sigma(\mathcal{R})$ and the set of all $\mathcal{R}$-recursive objects is $\mathbf{Rec}(\mathcal{R})$. Then $\mathbf{Rec}(\mathcal{R}) = \mathbb{O}(\mathbf{REC}(\mathcal{R}))$, so $\mathbf{Rec}(\mathcal{R})$ is a λ-set, because $\mathbf{REC}(\mathcal{R})$ is an elementary set of $\mathfrak{M}$-representable functionals. This gives us the following theorems.

4.3.12. THEOREM. (i) $a \in \mathbf{Rec}(\mathcal{R})$ & $a \approx b \Rightarrow b \in \mathbf{Rec}(\mathcal{R})$.

(ii) $a \in \mathbf{Rec}_{\sigma\to\tau}(\mathcal{R})$ & $b \in \mathbf{Rec}_\sigma(\mathcal{R}) \Rightarrow ab \in \mathbf{Rec}_\tau(\mathcal{R})$.

(iii) *There exists a quasi-k-family* $(\mathbf{k}_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}$ *in* $\mathfrak{M}$ *such that* $\mathbf{k}_{\sigma,\tau}$ *is* $\mathcal{R}$*-recursive for all* $\sigma, \tau \in \mathbb{T}$.

(iv) *There exists a quasi-s-family* $(\mathbf{s}_{\sigma,\tau,\upsilon})_{\sigma,\tau,\upsilon\in\mathbb{T}}$ *in* $\mathfrak{M}$ *such that* $\mathbf{s}_{\sigma,\tau,\upsilon}$ *is* $\mathcal{R}$-*recursive for all* $\sigma,\tau,\upsilon\in\mathbb{T}$. □

4.3.13. THEOREM. *Let* $[\![\,]\!]$ *be a quasi-*$\mathcal{T}(\mathbb{C})$*-interpretation in* $\mathfrak{M}$ *such that* $[\![\mathbf{C}]\!]$ *is* $\mathcal{R}$*-recursive for all* $\mathbf{C}\in|\mathbb{C}|$. *Then every definable object of* $(\mathfrak{M},[\![\,]\!])$ *is* $\mathcal{R}$*-recursive.* □

Because $fnl(m)=\mathsf{C}^0_m\in\mathbf{REC}(\mathcal{R})$ for every $m\in\omega$ we have that every natural number is an $\mathcal{R}$-recursive type-0 object.

Next we will prove the First Recursion Theorem for $\mathcal{R}$.

4.3.14. THEOREM (First Recursion Theorem).
For every $\sigma\in\mathbb{T}$ *there exists an* $\mathcal{R}$*-recursive object* $\mathbf{y}_\sigma\in\mathfrak{M}_{(\sigma\to\sigma)\to\sigma}$ *such that for all* $f\in\mathfrak{M}_{\sigma\to\sigma}$

$$\mathbf{y}_\sigma f\approx f(\mathbf{y}_\sigma f).\ \square$$

The proof of this theorem is established in 4.3.15-4.3.19.

Let $\sigma\in\mathbb{T}$ and define $\tau=0\to(\sigma\to\sigma)\to\sigma$.

4.3.15. DEFINITION. (i) Let $\psi_\sigma:\mathfrak{M}_0\times\mathfrak{M}_{\sigma\to\sigma}\times(\mathfrak{M})_\sigma\times\mathfrak{M}_\sigma\to\mathfrak{M}_0$ be defined by

$$\psi_\sigma(e,f,\vec{a},b)=fb\vec{a}.$$

(ii) Let $\chi_\sigma:\mathfrak{M}_0\times\mathfrak{M}_{\sigma\to\sigma}\times(\mathfrak{M})_\sigma\times(\mathfrak{M})_\sigma\to\mathfrak{M}_0$ be defined by

$$\chi_\sigma(e,f,\vec{a},\vec{c})=\mathcal{E}_\tau(e,e,f,\vec{c}).$$

(iii) Define $\varphi_\sigma=\mathsf{Sub}_{\tau,\sigma\to 0}[\psi_\sigma,\chi_\sigma]$. □

4.3.16. LEMMA. (i) $\varphi_\sigma:\mathfrak{M}_0\times\mathfrak{M}_{\sigma\to\sigma}\times(\mathfrak{M})_\sigma\to\mathfrak{M}_0$ *is an* $\mathcal{R}$*-recursive functional.*
(ii) *For all* $e\in\mathfrak{M}_0$, $f\in\mathfrak{M}_{\sigma\to\sigma}$, $b\in\mathfrak{M}_\sigma$

$$\forall\vec{c}\in(\mathfrak{M})_\sigma[\,b\vec{c}=\mathcal{E}_\tau(e,e,f,\vec{c})\,]\Rightarrow\forall\vec{a}\in(\mathfrak{M})_\sigma[\,\varphi_\sigma(e,f,\vec{a})=fb\vec{a}\,].$$

PROOF. (i) Easy.
(ii) Let $e\in\mathfrak{M}_0$, $f\in\mathfrak{M}_{\sigma\to\sigma}$ and $b\in\mathfrak{M}_\sigma$ be such that

$$\forall\vec{c}\in(\mathfrak{M})_\sigma[\,b\vec{c}=\mathcal{E}_\tau(e,e,f,\vec{c})\,].$$

Now let $\vec{a}\in(\mathfrak{M})_\sigma$. Then for all $\vec{c}\in(\mathfrak{M})_\sigma$

$$b\vec{c}=\chi_\sigma(e,f,\vec{a},\vec{c})$$

and so, by 4.3.15

$$\varphi_\sigma(e,f,\vec{a})=\psi_\sigma(e,f,\vec{a},b)=fb\vec{a}.\ \square$$

Let $e_0 \in \mathbf{Ix}(\tau)$ be such that $\varphi_\sigma = \{e_0\}^\tau$.

4.3.17. DEFINITION. Let $\Phi_\sigma : \mathfrak{M}_{\sigma\to\sigma} \times (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ be defined by

$$\Phi_\sigma(f, \vec{a}) = \varphi_\sigma(e_0, f, \vec{a}). \quad \square$$

4.3.18. LEMMA. (i) *Φ_σ is $\mathcal{R}$-recursive.*
(ii) *There exists an $\mathcal{R}$-recursive object $\mathbf{y}_\sigma \in \mathfrak{M}_{(\sigma\to\sigma)\to\sigma}$ with $\Phi_\sigma = fnl(\mathbf{y}_\sigma)$.*

PROOF. (i) Easy.
(ii) By (i). $\square$

4.3.19. LEMMA. *Let $\mathbf{y}_\sigma$ be an $\mathcal{R}$-recursive object with $\Phi_\sigma = fnl(\mathbf{y}_\sigma)$. Then for all $f \in \mathfrak{M}_{\sigma\to\sigma}$*

$$\mathbf{y}_\sigma f \approx f(\mathbf{y}_\sigma f).$$

PROOF. Let $f \in \mathfrak{M}_{\sigma\to\sigma}$. Define $b = \mathbf{y}_\sigma f \in \mathfrak{M}_\sigma$. Let $\vec{a} \in (\mathfrak{M})_\sigma$. Then for all $\vec{c} \in (\mathfrak{M})_\sigma$

$$\begin{aligned} b\vec{c} &= \mathbf{y}_\sigma f \vec{c} \\ &= \Phi_\sigma(f, \vec{c}) \\ &= \varphi_\sigma(e_0, f, \vec{c}) \\ &= \{e_0\}^\tau(e_0, f, \vec{c}) \\ &= \mathcal{E}_\tau(e_0, e_0, f, \vec{c}) \end{aligned}$$

because $\varphi_\sigma = \{e_0\}^\tau$ and $e_0 \in \mathbf{Ix}(\tau)$. So, by 4.3.16 $\varphi_\sigma(e_0, f, \vec{a}) = fb\vec{a} = f(\mathbf{y}_\sigma f)\vec{a}$. But also $\varphi_\sigma(e_0, f, \vec{a}) = \Phi_\sigma(f, \vec{a}) = \mathbf{y}_\sigma f\vec{a}$. And so $\mathbf{y}_\sigma f\vec{a} = f(\mathbf{y}_\sigma f)\vec{a}$. Therefore $\mathbf{y}_\sigma f \approx f(\mathbf{y}_\sigma f)$. $\square$

For any $\sigma \in \mathbb{T}$ take a fixed $\mathcal{R}$-recursive object $\mathbf{y}_\sigma$ of type $(\sigma\to\sigma)\to\sigma$ with the property that for all $f \in \mathfrak{M}_{\sigma\to\sigma}$ we have $\mathbf{y}_\sigma f \approx f(\mathbf{y}_\sigma f)$.

4.3.20. DEFINITION. Let $\sigma \in \mathbb{T}$, $f \in \mathfrak{M}_{\sigma\to\sigma}$, $a \in \mathfrak{M}_\sigma$. Then a is called a weak fixed point of f if $fa \approx a$. $\square$

4.3.21. THEOREM. *Let $\sigma \in \mathbb{T}$.*
(i) *Every $\mathcal{R}$-recursive $f \in \mathfrak{M}_{\sigma\to\sigma}$ has an $\mathcal{R}$-recursive weak fixed point.*
(ii) *Let $\varphi : \mathfrak{M}_\sigma \times (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ be $\mathcal{R}$-recursive. Then there exists an $\mathcal{R}$-recursive $a \in \mathfrak{M}_\sigma$ such that for all $\vec{b} \in (\mathfrak{M})_\sigma$*

$$a\vec{b} = \varphi(a, \vec{b}).$$

PROOF. (i) Let $f \in \mathfrak{M}_{\sigma\to\sigma}$ be $\mathcal{R}$-recursive. Then $\mathbf{y}_\sigma f$ is an $\mathcal{R}$-recursive weak fixed point of f.

(ii) Let $f \in \mathfrak{M}_{\sigma\to\sigma}$ $\mathfrak{M}$-represent φ. Then f is $\mathcal{R}$-recursive, and so, by (i), f has an $\mathcal{R}$-recursive weak fixed point $a \in \mathfrak{M}_\sigma$. Then for all $\vec{b} \in (\mathfrak{M})_\sigma$

$$a\vec{b} = fa\vec{b} = \varphi(a, \vec{b}). \ \Box$$

4.3.22. COROLLARY. (i) *Every $\mathcal{R}$-recursive function $\varphi : \mathfrak{M}_0 \to \mathfrak{M}_0$ has an $\mathcal{R}$-recursive fixed point.*

(ii) *$\mathfrak{M}_0$ contains an $\mathcal{R}$-recursive object a with $a \notin \omega$. So $\mathfrak{M}_0 \neq \omega$.*

PROOF. (i) By (i) of the previous theorem, taking $\sigma = 0$.

(ii) Let a be an $\mathcal{R}$-recursive fixed point of $\mathsf{S}^+ : \mathfrak{M}_0 \to \mathfrak{M}_0$. Then $a \notin \omega$, because $\forall m \in \omega[\, \mathsf{S}^+(m) \neq m\,]$. $\Box$

4.3.23. THEOREM. *There exists a quasi-$\lambda\mathsf{Y}(\mathcal{N})$-interpretation $[\![\,]\!]$ in $\mathfrak{M}$ such that $(\mathfrak{M}, [\![\,]\!])$ is a quasi-$\lambda\mathsf{Y}(\mathcal{N})$-model with $[\![\mathbf{c}_m]\!] = m$ for all $m \in \omega$ and such that every definable object of $(\mathfrak{M}, [\![\,]\!])$ is $\mathcal{R}$-recursive.*

PROOF. Because $\mathfrak{M}$ is a λ-structure, there exists a quasi-$\lambda\mathsf{Y}(\mathcal{N})$-interpretation $[\![\,]\!]$ in $\mathfrak{M}$ such that

(i) $[\![\mathbf{Y}_\sigma]\!] = \mathbf{y}_\sigma$
(ii) $[\![\mathbf{c}_m]\!] = m$
(iii) $[\![\mathbf{S}^+]\!]$ $\mathfrak{M}$-represents S^+
(iv) $[\![\boldsymbol{\Delta}]\!]$ $\mathfrak{M}$-represents Δ.

In order to prove that $(\mathfrak{M}, [\![\,]\!])$ is a quasi-$\lambda\mathsf{Y}(\mathcal{N})$-model, we only need to check the quasi-satisfaction of the δ-axioms of $\lambda\mathsf{Y}(\mathcal{N})$, because $\mathfrak{M}$ is quasi-extensional. First the γ-axioms. Let $\sigma \in \mathbb{T}$ and $x \in Var_{\sigma\to\sigma}$. Let $\rho \in Val(\mathfrak{M})$. Then for all $f \in \mathfrak{M}_{\sigma\to\sigma}$

$$\begin{array}{rcl} [\![\mathbf{Y}_\sigma]\!]f &=& \mathbf{y}_\sigma f \\ &\approx& f(\mathbf{y}_\sigma f) \\ &=& [\![x(\mathbf{Y}_\sigma x)]\!]_{\rho(x:=f)} \\ &\approx& [\![\lambda x.x(\mathbf{Y}_\sigma x)]\!]f \end{array}$$

so $[\![\mathbf{Y}_\sigma]\!] \approx [\![\lambda x.x(\mathbf{Y}_\sigma x)]\!]$, and so $\mathfrak{M}, [\![\,]\!] \models_q \mathbf{Y}_\sigma = \lambda x.x(\mathbf{Y}_\sigma x)$.

Now the α-axioms of $\lambda\mathsf{Y}(\mathcal{N})$. There are three kinds of α-axioms.

(i) Let $m \in \omega$. Then $(\mathbf{S}^+\mathbf{c}_m = \mathbf{c}_{m+1})$ is an α-axiom. We have

$$\begin{array}{rcl} [\![\mathbf{S}^+\mathbf{c}_m]\!] &=& [\![\mathbf{S}^+]\!][\![\mathbf{c}_m]\!] \\ &=& [\![\mathbf{S}^+]\!]m \\ &=& \mathsf{S}^+(m) \\ &=& m+1 \\ &=& [\![\mathbf{c}_{m+1}]\!] \end{array}$$

because $[\![\mathbf{S}^+]\!]$ $\mathfrak{M}$-represents S^+. So

$$\mathfrak{M}, [\![\,]\!] \models_q \mathbf{S}^+\mathbf{c}_m = \mathbf{c}_{m+1}.$$

(ii) Let $m \in \omega$. Then $(\mathbf{\Delta c}_m\mathbf{c}_m = \mathbf{true}_0)$ is an α-axiom of $\lambda\mathsf{Y}(\mathcal{N})$. For all $a, b \in \mathfrak{M}_0$

$$\begin{aligned}
[\![\mathbf{\Delta c}_m\mathbf{c}_m]\!]ab &= [\![\mathbf{\Delta}]\!][\![\mathbf{c}_m]\!][\![\mathbf{c}_m]\!]ab \\
&= [\![\mathbf{\Delta}]\!]mmab \\
&= \Delta(m, m, a, b) \\
&= a \\
&= [\![\mathbf{true}_0]\!]ab
\end{aligned}$$

so $[\![\mathbf{\Delta c}_m\mathbf{c}_m]\!] \approx [\![\mathbf{true}_0]\!]$, and so

$$\mathfrak{M}, [\![\,]\!] \models_q \mathbf{\Delta c}_m\mathbf{c}_m = \mathbf{true}_0.$$

(iii) Let $m, n \in \omega$ be such that $m \neq n$. Then $(\mathbf{\Delta c}_m\mathbf{c}_n = \mathbf{false}_0)$ is an α-axiom of $\lambda\mathsf{Y}(\mathcal{N})$. The proof of the quasi-satisfaction of this α-axiom is similar to (ii).

Then it follows by 2.2.14 that $(\mathfrak{M}, [\![\,]\!])$ is a quasi-$\lambda\mathsf{Y}\eta(\mathcal{N})$-model and a quasi-$\lambda\mathsf{Y}(\mathcal{N})$-model. This proves (i). As for (ii), this follows immediately by 4.3.13, because for every constant $\mathbf{C}$ of $\lambda\mathsf{Y}(\mathcal{N})$ we have that the object $[\![\mathbf{C}]\!]$ is $\mathcal{R}$-recursive. □

The previous theorem is one of our main reasons for working with quasi-models. For quasi-extensional $\mathfrak{M}$ this is the best we can do.

In the following $[\![\,]\!]$ is a fixed quasi-$\lambda\mathsf{Y}(\mathcal{N})$-interpretation in $\mathfrak{M}$ such that

- $(\mathfrak{M}, [\![\,]\!])$ is a quasi-$\lambda\mathsf{Y}(\mathcal{N})$-model
- $[\![\mathbf{c}_m]\!] = m$ for all $m \in \omega$
- every definable object of $(\mathfrak{M}, [\![\,]\!])$ is $\mathcal{R}$-recursive.

We are going to show that every recursive function on ω has an $\mathcal{R}$-recursive extension. These extensions will have the property that they are ω-invariant. We will first define this notion of ω-invariance.

4.3.24. Definition. Let $k \in \omega$ and let $\varphi : (\mathfrak{M}_0)^k \to \mathfrak{M}_0$. Then φ is ω-invariant if for all $\vec{m} \in \omega^k$ we have $\varphi(\vec{m}) \in \omega$. □

4.3.25. Theorem (Extension Theorem).
For every recursive function $f : \omega^k \to \omega$ there exists an ω-invariant $\mathcal{R}$-recursive function $\varphi : (\mathfrak{M}_0)^k \to \mathfrak{M}_0$ such that

$$\forall \vec{m} \in \omega^k [\, \varphi(\vec{m}) = f(\vec{m}) \,].$$

PROOF. Let $f : \omega^k \to \omega$ be recursive. Then f is $\lambda\mathsf{Y}(\mathcal{N})$-computable w.r.t. $\mathbf{c}$. Let $F \in \mathcal{T}_{t(k)}(\lambda\mathsf{Y}(\mathcal{N}))^0$ $\lambda\mathsf{Y}(\mathcal{N})$-compute f w.r.t. $\mathbf{c}$. Then $[\![F]\!]$ is an $\mathcal{R}$-recursive object of type $t(k)$ in $\mathfrak{M}$. Define $\varphi = fnl([\![F]\!])$. Then $\varphi : (\mathfrak{M}_0)^k \to \mathfrak{M}_0$ is $\mathcal{R}$-recursive and for all $\vec{m} \in \omega^k$

$$\begin{aligned}\varphi(\vec{m}) &= [\![F]\!]\vec{m}\\ &= [\![F]\!][\![\mathbf{c}_{m_1}]\!]\ldots[\![\mathbf{c}_{m_k}]\!]\\ &= [\![F\mathbf{c}_{m_1}\ldots\mathbf{c}_{m_k}]\!]\\ &= [\![\mathbf{c}_{f(\vec{m})}]\!]\\ &= f(\vec{m}).\end{aligned}$$

Obviously, φ is ω-invariant. □

4.3.26. THEOREM (First Parameter Theorem).
Let $m \in \omega$, $\sigma \in \mathbb{T}$. Then there exists a recursive function $S^m_\sigma : \omega^{m+1} \to \omega$ such that for all $e \in \mathbf{Ix}(t(m) * \sigma)$, $\vec{a} \in \omega^m$, $\vec{b} \in (\mathfrak{M})_\sigma$
(i) $S^m_\sigma(e, \vec{a}) \in \mathbf{Ix}(\sigma)$
(ii) $\{S^m_\sigma(e, \vec{a})\}^\sigma(\vec{b}) = \{e\}^{t(m)*\sigma}(\vec{a}, \vec{b})$.

PROOF. First note that there exists a recursive function $g : \omega \to \omega$ such that for all $e \in \mathbf{Ix}(t(m) * \sigma)$, $\vec{a} \in (\mathfrak{M}_0)^m$ and $\vec{b} \in (\mathfrak{M})_\sigma$ we have $g(e) \in \mathbf{Ix}(\sigma * t(m))$ and

$$\{e\}^{t(m)*\sigma}(\vec{a}, \vec{b}) = \{g(e)\}^{\sigma * t(m)}(\vec{b}, \vec{a}).$$

Then for all $e \in \mathbf{Ix}(t(m) * \sigma)$, $\vec{a} \in \omega^m$, $\vec{b} \in (\mathfrak{M})_\sigma$

$$\begin{aligned}\{e\}^{t(m)*\sigma}(\vec{a}, \vec{b}) &= \{g(e)\}^{\sigma * t(m)}(\vec{b}, \vec{a})\\ &= \{g(e)\}^{\sigma * t(m)}(\vec{b}, \mathsf{C}^\sigma_{a_1}(\vec{b}), \ldots, \mathsf{C}^\sigma_{a_m}(\vec{b}))\\ &= \mathsf{Comp}_{\sigma,m}[\{g(e)\}^{\sigma * t(m)}, \mathsf{C}^\sigma_{a_1}, \ldots, \mathsf{C}^\sigma_{a_m}](\vec{b})\\ &= \mathsf{Comp}_{\sigma,m}[\{g(e)\}^{\sigma * t(m)}, \{\mathbf{con}(\#\sigma, a_1)\}^\sigma, \ldots, \{\mathbf{con}(\#\sigma, a_m)\}^\sigma](\vec{b})\\ &= \{\mathbf{comp}(\#\sigma, m, < g(e), \mathbf{con}(\#\sigma, a_1), \ldots, \mathbf{con}(\#\sigma, a_m) >)\}^\sigma(\vec{b}).\end{aligned}$$

Now define $S^m_\sigma : \omega^{m+1} \to \omega$ by

$$S^m_\sigma(e, \vec{a}) = \mathbf{comp}(\#\sigma, m, < g(e), \mathbf{con}(\#\sigma, a_1), \ldots, \mathbf{con}(\#\sigma, a_m) >).$$

Then S^m_σ has the required properties. □

4.3.27. THEOREM (Second Parameter Theorem).
Let $m \in \omega$, $\sigma \in \mathbb{T}$. Then there exists an ω-invariant $\mathcal{R}$-recursive function T^m_σ : $(\mathfrak{M}_0)^{m+1} \to \mathfrak{M}_0$ such that for all $e \in \mathbf{Ix}(t(m) * \sigma)$, $\vec{a} \in \omega^m$, $\vec{b} \in (\mathfrak{M})_\sigma$

(i) $T^m_\sigma(e, \vec{a}) \in \mathbf{Ix}(\sigma)$
(ii) $\{T^m_\sigma(e, \vec{a})\}^\sigma(\vec{b}) = \{e\}^{t(m)*\sigma}(\vec{a}, \vec{b})$.

PROOF. Let T^m_σ be an ω-invariant $\mathcal{R}$-recursive extension of S^m_σ. Then T^m_σ has the required properties. □

4.3.28. THEOREM (Second Recursion Theorem).
Let $\sigma \in \mathbb{T}$ and let $\varphi : \mathfrak{M}_0 \times (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ be $\mathcal{R}$-recursive. Then there exists $e \in \mathbf{Ix}(\sigma)$ such that for all $\vec{a} \in (\mathfrak{M})_\sigma$

$$\{e\}^\sigma(\vec{a}) = \varphi(e, \vec{a}).$$

PROOF. Define $\psi : \mathfrak{M}_0 \times (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ by

$$\psi(m, \vec{a}) = \varphi(T^1_\sigma(m, m), \vec{a}).$$

Using closure of $\mathbf{REC}(\mathcal{R})$ under manipulation and composition it follows that ψ is $\mathcal{R}$-recursive. So there exists $d \in \mathbf{Ix}(0 \to \sigma)$ with $\psi = \{d\}^{0\to\sigma}$. Define $e = T^1_\sigma(d, d)$. Then, by the Second Parameter Theorem, $e \in \mathbf{Ix}(\sigma)$ and for all $\vec{a} \in (\mathfrak{M})_\sigma$

$$\begin{aligned} \{e\}^\sigma(\vec{a}) &= \{T^1_\sigma(d, d)\}^\sigma(\vec{a}) \\ &= \{d\}^{0\to\sigma}(d, \vec{a}) \\ &= \psi(d, \vec{a}) \\ &= \varphi(T^1_\sigma(d, d), \vec{a}) \\ &= \varphi(e, \vec{a}). \ \square \end{aligned}$$

4.4. Recursion theories and quasi-$\lambda\mathsf{E}(\mathcal{N})$-models.

In the previous section we used the First Recursion Theorem to show that every ω-structure that carries a recursion theory can be made into a quasi-$\lambda\mathsf{Y}(\mathcal{N})$-model. In this section we will show, using the Second Recursion Theorem, that every ω-structure carrying a recursion theory can be made into a quasi-$\lambda\mathsf{E}(\mathcal{N})$-model. We will also prove an inverse of this result. Together these two results provide a characterization of the special ω-structures in terms of $\lambda\mathsf{E}(\mathcal{N})$.

Let $\mathfrak{M}$ be an ω-structure. Let $\mathcal{R}$ be a recursion theory on $\mathfrak{M}$. Fix recursive **Ix**, **type**, **ap**, **sub**, **man** and **enum** as described in section 4.3. Also fix $\mathcal{R}$-recursive functions S^+ and Δ as described in axioms (ii) and (iii) of the notion of recursion theory. And finally fix a quasi-k-family $(\mathbf{k}_{\sigma,\tau})_{\sigma,\tau\in\mathbb{T}}$ and a quasi-s-family $(\mathbf{s}_{\sigma,\tau,\upsilon})_{\sigma,\tau,\upsilon\in\mathbb{T}}$ in $\mathfrak{M}$.

Let $obj : \mathbb{F}(\mathfrak{M}) \to |\mathfrak{M}|$ be a mapping that chooses representatives for $\mathfrak{M}$-representable functionals. So for every $\varphi \in \mathbb{F}(\mathfrak{M})$ $obj(\varphi)$ is an object of $\mathfrak{M}$ that $\mathfrak{M}$-represents φ.

4.4.1. LEMMA. (i) $\varphi(\vec{a}) = obj(\varphi)\vec{a}$.
(ii) $fnl(obj(\varphi)) = \varphi$.
(iii) $obj(fnl(a)) \approx a$.

PROOF. Simple. □

We see that obj is a right-inverse of fnl. Note that, in general, for quasi-extensional $\mathfrak{M}$, we need the Axiom of Choice in order to introduce obj. But also note that we do not need the Axiom of Choice for extensional $\mathfrak{M}$, because in that case fnl is bijective and we can take $obj = fnl^{-1}$.

4.4.2. DEFINITION. Let $e \in \omega$, $\sigma \in \mathbb{T}$.
(i) $[e]^\sigma = obj(\{e\}^\sigma)$.
(ii) e is called an $\mathcal{R}$-index for $a \in \mathfrak{M}_\sigma$ if $e \in \mathbf{Ix}(\sigma)$ and $\{e\}^\sigma = fnl(a)$. □

4.4.3. LEMMA. *Let $e \in \omega$, $\sigma \in \mathbb{T}$, $a \in \mathfrak{M}_\sigma$. Then e is an $\mathcal{R}$-index for a if and only if $e \in \mathbf{Ix}(\sigma)$ and $[e]^\sigma \approx a$.*

PROOF. Simple. □

4.4.4. PROPOSITION. (i) *There exists an ω-invariant $\mathcal{R}$-recursive $\mathbf{k} : (\mathfrak{M}_0)^2 \to \mathfrak{M}_0$ such that for all $\sigma, \tau \in \mathbb{T}$ we have that $\mathbf{k}(\#\sigma, \#\tau)$ is an $\mathcal{R}$-index for $\mathbf{k}_{\sigma,\tau}$.*
(ii) *There exists an ω-invariant $\mathcal{R}$-recursive $\mathbf{s} : (\mathfrak{M}_0)^3 \to \mathfrak{M}_0$ such that for all $\sigma, \tau, \upsilon \in \mathbb{T}$ we have that $\mathbf{s}(\#\sigma, \#\tau, \#\upsilon)$ is an $\mathcal{R}$-index for $\mathbf{s}_{\sigma,\tau,\upsilon}$.*
(iii) *There exists an ω-invariant $\mathcal{R}$-recursive $\mathbf{app} : (\mathfrak{M}_0)^2 \to \mathfrak{M}_0$ such that for all $\sigma, \tau \in \mathbb{T}$ and all $e_1 \in \mathbf{Ix}(\sigma \to \tau)$, $e_2 \in \mathbf{Ix}(\sigma)$ we have $\mathbf{app}(\#\sigma, \#\tau) \in \mathbf{Ix}(\tau)$ and*

$$[e_1]^{\sigma\to\tau}[e_2]^\sigma \approx [\mathbf{app}(e_1, e_2)]^\tau.$$

(iv) *There exists an ω-invariant $\mathcal{R}$-recursive $\mathbf{nat} : \mathfrak{M}_0 \to \mathfrak{M}_0$ such that for all $m \in \omega$ $\mathbf{nat}(m)$ is an $\mathcal{R}$-index for m.*

PROOF. (i) By 4.2.13(ii) there exists a recursive function $\mathbf{k}_0 : \omega^2 \to \omega$ such that for all $\sigma, \tau \in \mathbb{T}$ $\mathbf{k}_0(\#\sigma, \#\tau)$ is an $\mathcal{R}$-index for $\mathbf{k}_{\sigma,\tau}$. Let $\mathbf{k}$ be an ω-invariant $\mathcal{R}$-recursive extension of $\mathbf{k}_0$.
(ii) By 4.2.13(iii) there exists a recursive function $\mathbf{s}_0 : \omega^3 \to \omega$ such that for all $\sigma, \tau, \upsilon \in \mathbb{T}$ $\mathbf{s}_0(\#\sigma, \#\tau, \#\upsilon)$ is an $\mathcal{R}$-index for $\mathbf{s}_{\sigma,\tau,\upsilon}$. Let $\mathbf{s}$ be an ω-invariant $\mathcal{R}$-recursive extension of $\mathbf{s}_0$.

(iii) By 4.2.13(i) there exists a recursive function $\mathbf{app}_0 : \omega^2 \to \omega$ such that for all $\sigma, \tau \in \mathbb{T}$ and all $e_1 \in \mathbf{Ix}(\sigma \to \tau)$, $e_2 \in \mathbf{Ix}(\sigma)$ we have $\mathbf{app}_0(e_1, e_2) \in \mathbf{Ix}(\tau)$ and

$$[e_1]^{\sigma\to\tau}[e_2]^{\sigma} \approx [\mathbf{app}_0(e_1, e_2)]^{\tau}.$$

Let $\mathbf{app}$ be an ω-invariant $\mathcal{R}$-recursive extension of $\mathbf{app}_0$.

(iv) Define $\mathbf{nat}_0 : \omega \to \omega$ by $\mathbf{nat}_0(m) = \mathbf{con}(\#0, m)$. Then $\mathbf{nat}_0$ is recursive. Now let $\mathbf{nat}$ be an ω-invariant $\mathcal{R}$-recursive extension of $\mathbf{nat}_0$. $\square$

From now on let $\mathbf{k}$, $\mathbf{s}$, $\mathbf{app}$ and $\mathbf{nat}$ be fixed ω-invariant $\mathcal{R}$-recursive functions with the properties as described in the above proposition. Also fix $s^+ \in \mathbf{Ix}(0\to0)$ and $\delta \in \mathbf{Ix}(t(4))$ such that $\mathsf{S}^+ = \{s^+\}^{0\to0}$ and $\Delta = \{\delta\}^{t(4)}$.

Let $\mathbb{C}$ be such that $\mathcal{T}(\lambda\mathsf{E}(\mathcal{N})) = \mathcal{T}(\mathbb{C})$. Then we also have the set $\mathbf{CL}(\mathbb{C}^*)$ of combinatory terms with constants from $\mathbb{C}$ and with extra constants $\mathsf{K}_{\sigma,\tau}$ and $\mathsf{S}_{\sigma,\tau,\upsilon}$.

In the following we are going to define, for every $e \in \mathfrak{M}_0$ and every $P \in \mathbf{CL}(\mathbb{C}^*)^0$, an element $\mathbf{ind}_e(P)$ of $\mathfrak{M}_0$. After that we will construct an $e_0 \in \mathbf{Ix}(0\to0)$ and a $\mathbf{CL}(\mathbb{C}^*)$-interpretation $(\ )$ in $\mathfrak{M}$ such that $\mathbf{ind}_{e_0}(P)$ is an $\mathcal{R}$-index for (P) for every $P \in \mathbf{CL}(\mathbb{C}^*)^0$.

4.4.5. Definition. Let $e \in \mathfrak{M}_0$. Define the function $\mathbf{ind}_e : \mathbf{CL}(\mathbb{C}^*)^0 \to \mathfrak{M}_0$ by induction on the structure of the terms in $\mathbf{CL}(\mathbb{C}^*)^0$ as follows.

(i) $\mathbf{ind}_e(\mathsf{K}_{\sigma,\tau}) = \mathbf{k}(\#\sigma, \#\tau)$.
(ii) $\mathbf{ind}_e(\mathsf{S}_{\sigma,\tau,\upsilon}) = \mathbf{s}(\#\sigma, \#\tau, \#\upsilon)$.
(iii) $\mathbf{ind}_e(\mathbf{c}_m) = \mathbf{nat}(m)$.
(iv) $\mathbf{ind}_e(\mathbf{S}^+) = s^+$.
(v) $\mathbf{ind}_e(\boldsymbol{\Delta}) = \delta$.
(vi) $\mathbf{ind}_e(\mathbf{E}_\sigma) = \mathcal{E}_{0\to0}(e, \#\sigma)$.
(vii) $\mathbf{ind}_e(PQ) = \mathbf{app}(\mathbf{ind}_e(P), \mathbf{ind}_e(Q))$. $\square$

We are going to prove that $\mathbf{ind}_e(P)$ can be represented by an $\mathcal{R}$-recursive function $\mathbf{Ind} : (\mathfrak{M})^2 \to \mathfrak{M}_0$. For this we need a coding function $\# : \mathbf{CL}(\mathbb{C}^*)^0 \to \omega$ and we need to prove closure of $\mathbf{REC}(\mathcal{R})$ under a certain principle of definition by cases.

4.4.6. Definition. For every $P \in \mathbf{CL}(\mathbb{C}^*)^0$ define $\#(P) \in \omega$ by induction on the structure of P as follows.

(i) $\#(\mathsf{K}_{\sigma,\tau}) =< 0, \#\sigma, \#\tau >$.
(ii) $\#(\mathsf{S}_{\sigma,\tau,\upsilon}) =< 1, \#\sigma, \#\tau, \#\upsilon >$.
(iii) $\#(\mathbf{c}_m) =< 2, m >$.
(iv) $\#(\mathbf{S}^+) =< 3 >$.
(v) $\#(\boldsymbol{\Delta}) =< 4 >$.

(vi) $\#(\mathbf{E}_\sigma) = <5, \#\sigma>$.
(vii) $\#(PQ) = <6, \#(P), \#(Q)>$. □

We will now define the operators on $\mathbb{F}(\mathfrak{M})$ that correspond to the abovementioned definition by cases. The purpose of these operators will become clear in theorem 4.4.9.

4.4.7. DEFINITION. Let $\sigma \in \mathbb{T}$. For every $n \in \omega$ we define an operator

$$\mathsf{Case}_{\sigma,n} : \mathbb{F}_\sigma(\mathfrak{M})^{n+2} \to \mathbb{F}_\sigma(\mathfrak{M})$$

by induction on n as follows.

(i) Let $\varphi, \psi_0 : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ be $\mathfrak{M}$-representable. Then $\mathsf{Case}_{\sigma,0}[\varphi, \psi_0]$ is the functional $\chi : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ defined by

$$\chi(\vec{a}) = \Delta(\varphi(\vec{a}), 0, \psi_0(\vec{a}), 0).$$

(ii) Let $n \in \omega$. Let $\varphi, \psi_0, \ldots, \psi_{n+1} : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ be $\mathfrak{M}$-representable. First let

$$\chi_0 = \mathsf{Case}_{\sigma,n}[\varphi, \psi_0, \ldots, \psi_n].$$

Then $\mathsf{Case}_{\sigma,n+1}[\varphi, \psi_0, \ldots, \psi_{n+1}]$ is the functional $\chi : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ defined by

$$\chi(\vec{a}) = \Delta(\varphi(\vec{a}), n+1, \psi_{n+1}(\vec{a}), \chi_0(\vec{a})). \ \square$$

4.4.8. LEMMA. *Let $\sigma \in \mathbb{T}$, $n \in \omega$ and let $\varphi, \psi_0, \ldots, \psi_n$ be $\mathcal{R}$-recursive type-σ functionals. Then $\mathsf{Case}_{\sigma,n}[\varphi, \psi_0, \ldots, \psi_n]$ is also $\mathcal{R}$-recursive.*

PROOF. By induction on n. □

4.4.9. THEOREM. *Let $\sigma \in \mathbb{T}$, $n \in \omega$ and let $\varphi, \psi_0, \ldots, \psi_n$ be $\mathcal{R}$-recursive. Then there exists an $\mathcal{R}$-recursive functional $\chi : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ such that for all $\vec{a} \in (\mathfrak{M})_\sigma$ and $j \in \{0, 1, \ldots, n\}$*

$$\varphi(\vec{a}) = j \Rightarrow \chi(\vec{a}) = \psi_j(\vec{a}).$$

PROOF. Take $\chi = \mathsf{Case}_{\sigma,n}[\varphi, \psi_0, \ldots, \psi_n]$. □

4.4.10. PROPOSITION. *There exists an $\mathcal{R}$-recursive function $\mathbf{Ind} : (\mathfrak{M}_0)^2 \to \mathfrak{M}_0$ such that for all $e \in \mathfrak{M}_0$ and $P \in \mathbf{CL}(\mathbb{C}^*)^0$*

$$\mathbf{Ind}(e, \#(P)) = \mathbf{ind}_e(P).$$

PROOF. For every $j \in \omega$ let $[\,]_j$ be an ω-invariant $\mathcal{R}$-recursive extension of the j-th decoding function $(\,)_j : \omega \to \omega$. By the previous theorem (taking $n = 6$) there exists an $\mathcal{R}$-recursive function $\varphi : (\mathfrak{M}_0)^3 \to \mathfrak{M}_0$ such that for all $i, e, p \in \mathfrak{M}_0$

(i) $\varphi(i, e, p) = \mathbf{k}([p]_1, [p]_2)$ if $[p]_0 = 0$

(ii) $\varphi(i,e,p) = \mathbf{s}([p]_1,[p]_2,[p]_3)$ if $[p]_0 = 1$
(iii) $\varphi(i,e,p) = \mathbf{nat}([p]_1)$ if $[p]_0 = 2$
(iv) $\varphi(i,e,p) = s^+$ if $[p]_0 = 3$
(v) $\varphi(i,e,p) = \delta$ if $[p]_0 = 4$
(vi) $\varphi(i,e,p) = \mathcal{E}_{0\to 0}(e,[p]_1)$ if $[p]_0 = 5$
(vii) $\varphi(i,e,p) = \mathbf{app}(\mathcal{E}_{t(2)}(i,e,[p]_1), \mathcal{E}_{t(2)}(i,e,[p]_2))$ if $[p]_0 = 6$.

By the Second Recursion Theorem there exists $i_0 \in \mathbf{Ix}(t(2))$ such that for all $e,p \in \mathfrak{M}_0$

$$\{i_0\}^{t(2)}(e,p) = \varphi(i_0,e,p).$$

Define $\mathbf{Ind} = \{i_0\}^{t(2)} : (\mathfrak{M}_0)^2 \to \mathfrak{M}_0$. Then $\mathbf{Ind}$ is $\mathcal{R}$-recursive and for all $e,p \in \mathfrak{M}_0$

$$\mathbf{Ind}(e,p) = \varphi(i_0,e,p).$$

By induction on the structure of P it will now be shown that for every $P \in \mathbf{CL}(\mathbb{C}^*)^0$ and all $e \in \mathfrak{M}_0$

$$\mathbf{Ind}(e,\#(P)) = \mathbf{ind}_e(P).$$

(i) Using the definition of $\mathbf{ind}$ and the properties of φ described above it follows easily that for every $\mathbf{C} \in |\mathbb{C}|$ we have $\mathbf{Ind}(e,\#(\mathbf{C})) = \mathbf{ind}_e(\mathbf{C})$.

(ii) The induction step:

$$\begin{aligned}
\mathbf{Ind}(e,\#(PQ)) &= \varphi(i_0,e,\#(PQ)) \\
&= \mathbf{app}(\mathcal{E}_{t(2)}(i_0,e,\#(P)), \mathcal{E}_{t(2)}(i_0,e,\#(Q)) \\
&= \mathbf{app}(\{i_0\}^{t(2)}(e,\#(P)), \{i_0\}^{t(2)}(e,\#(Q))) \\
&= \mathbf{app}(\mathbf{Ind}(e,\#(P)), \mathbf{Ind}(e,\#(Q)) \\
&= \mathbf{app}(\mathbf{ind}_e(P), \mathbf{ind}_e(Q)) \\
&= \mathbf{ind}_e(PQ). \ \square
\end{aligned}$$

4.4.11. LEMMA. *There exists an ω-invariant $\mathcal{R}$-recursive function $\mathbf{cl} : \mathfrak{M}_0 \to \mathfrak{M}_0$ such that for all $M \in \mathcal{T}(\lambda\mathsf{E}(\mathcal{N}))^0$*

$$\mathbf{cl}(\#M) = \#(M_{CL}).$$

PROOF. Recall that $CL : \mathcal{T}(\lambda\mathsf{E}(\mathcal{N})) \to \mathbf{CL}(\mathbb{C}^*)$ is the transformation defined in section 1.6. It is easy to see that there exists a recursive function $\mathbf{cl}_0 : \omega \to \omega$ such that for all $M \in \mathcal{T}(\lambda\mathsf{E}(\mathcal{N}))^0$ we have $\mathbf{cl}_0(\#M) = \#(M_{CL})$. Now let $\mathbf{cl}$ be an ω-invariant $\mathcal{R}$-recursive extension of $\mathbf{cl}_0$. $\square$

4.4.12. PROPOSITION. *There exists an ω-invariant $\mathcal{R}$-recursive $g : (\mathfrak{M}_0)^2 \to \mathfrak{M}_0$ such that for all $\sigma \in \mathbb{T}$, $e \in \omega$, $M \in \mathcal{T}(\lambda\mathsf{E}(\mathcal{N}))^0$ and $\vec{a} \in (\mathfrak{M})_\sigma$ we have $g(e,\#\sigma) \in \mathbf{Ix}(0 \to \sigma)$ and*

$$\{g(e,\#\sigma)\}^{0\to\sigma}(\#M,\vec{a}) = \mathcal{E}_\sigma(\mathbf{ind}_e(M_{CL}),\vec{a}).$$

PROOF. It is easy to see that there exists a recursive $f_0 : \omega \to \omega$ such that for all $\sigma \in \mathbb{T}$, $e, m \in \mathfrak{M}_0$, $\vec{a} \in (\mathfrak{M})_\sigma$ we have $f_0(\#\sigma) \in \mathbf{Ix}(t(2) * \sigma)$ and

$$\{f_0(\#\sigma)\}^{t(2)*\sigma}(e, m, \vec{a}) = \mathcal{E}_\sigma(\mathbf{Ind}(e, \mathbf{cl}(m), \vec{a}).$$

Let $f : \mathfrak{M}_0 \to \mathfrak{M}_0$ be an ω-invariant $\mathcal{R}$-recursive extension of f_0, and define $g : (\mathfrak{M}_0)^2 \to \mathfrak{M}_0$ by

$$g(e, u) = T^1_{0\to\sigma}(f(u), e).$$

Then g is ω-invariant and $\mathcal{R}$-recursive, because $T^1_{0\to\sigma}$ and f are ω-invariant and $\mathcal{R}$-recursive. Now let $\sigma \in \mathbb{T}$, $e \in \omega$, $M \in \mathcal{T}_\sigma(\lambda\mathsf{E}(\mathcal{N}))^0$ and $\vec{a} \in (\mathfrak{M})_\sigma$. Then

$$g(e, \#\sigma) = T^1_{0\to\sigma}(f(\#\sigma), e) \in \mathbf{Ix}(0\to\sigma)$$

because $f(\#\sigma) \in \mathbf{Ix}(t(2) * \sigma)$, and further

$$\begin{array}{rcl}
\{g(e, \#\sigma)\}^{0\to\sigma}(\#M, \vec{a}) & = & \{T^1_{0\to\sigma}(f(\#\sigma), e)\}^{0\to\sigma}(\#M, \vec{a}) \\
& = & \{f(\#\sigma)\}^{t(2)*\sigma}(e, \#M, \vec{a}) \\
& = & \{f_0(\#\sigma)\}^{t(2)*\sigma}(e, \#M, \vec{a}) \\
& = & \mathcal{E}_\sigma(\mathbf{Ind}(e, \mathbf{cl}(\#M)), \vec{a}) \\
& = & \mathcal{E}_\sigma(\mathbf{Ind}(e, \#(M_{CL})), \vec{a}) \\
& = & \mathcal{E}_\sigma(\mathbf{ind}_e(M_{CL}), \vec{a}). \ \Box
\end{array}$$

4.4.13. PROPOSITION. *There exists $e_0 \in \mathbf{Ix}(0\to 0)$ such that for all $u \in \mathfrak{M}_0$*

$$\{e_0\}^{0\to 0}(u) = g(e_0, u).$$

PROOF. By the Second Recursion Theorem. $\Box$

From now on we let $e_0 \in \mathbf{Ix}(0\to 0)$ be such that for all $u \in \mathfrak{M}_0$

$$\{e_0\}^{0\to 0}(u) = g(e_0, u).$$

4.4.14. LEMMA. (i) *$\{e_0\}^{0\to 0}$ is ω-invariant.*

(ii) *$\mathbf{ind}_{e_0}(\mathbf{E}_\sigma) = g(e_0, \#\sigma)$ for all $\sigma \in \mathbb{T}$.*

(iii) *For all $\sigma \in \mathbb{T}$ and $P \in \mathbf{CL}_\sigma(\mathbb{C}^*)^0$*

$$\mathbf{ind}_{e_0}(P) \in \mathbf{Ix}(\sigma).$$

PROOF. (i) By the ω-invariance of g.

(ii) We have $e_0 \in \mathbf{Ix}(0\to 0)$, so for all $\sigma \in \mathbb{T}$

$$\begin{array}{rcl}
\mathbf{ind}_{e_0}(\mathbf{E}_\sigma) & = & \mathcal{E}_{0\to 0}(e_0, \#\sigma) \\
& = & \{e_0\}^{0\to 0}(\#\sigma) \\
& = & g(e_0, \#\sigma).
\end{array}$$

(iii) By induction on the structure of P

(1) $\mathbf{ind}_{e_0}(\mathsf{K}_{\sigma,\tau}) = \mathbf{k}(\#\sigma\#\tau) \in \mathbf{Ix}(\sigma\to\tau\to\sigma)$.

(2) $\mathbf{ind}_{e_0}(\mathsf{S}_{\sigma,\tau,\upsilon}) = \mathbf{s}(\#\sigma,\#\tau,\#\upsilon) \in \mathbf{Ix}((\sigma\to\tau\to\upsilon)\to(\sigma\to\tau)\to\sigma\to\upsilon)$.

(3) $\mathbf{ind}_{e_0}(\mathbf{c}_m) = \mathbf{nat}(m) \in \mathbf{Ix}(0)$.

(4) $\mathbf{ind}_{e_0}(\mathbf{S}^+) = s^+ \in \mathbf{Ix}(0\to 0)$.

(5) $\mathbf{ind}_{e_0}(\mathbf{\Delta}) = \delta \in \mathbf{Ix}(t(4))$.

(6) $\mathbf{ind}_{e_0}(\mathbf{E}_\sigma) = g(e_0,\#\sigma) \in \mathbf{Ix}(0\to\sigma)$.

(7) Let $P \in \mathbf{CL}_{\sigma\to\tau}(\mathbb{C}^*)^0$ and $Q \in \mathbf{CL}_\sigma(\mathbb{C}^*)^0$. Then by induction $\mathbf{ind}_{e_0}(P) \in \mathbf{Ix}(\sigma\to\tau)$ and $\mathbf{ind}_{e_0}(Q) \in \mathbf{Ix}(\sigma)$, and so

$$\mathbf{ind}_{e_0}(PQ) = \mathbf{app}(\mathbf{ind}_{e_0}(P), \mathbf{ind}_{e_0}(Q)) \in \mathbf{Ix}(\tau).\ \Box$$

4.4.15. Definition. Let ( ) be the $\mathbf{CL}(\mathbb{C}^*)$-interpretation in $\mathfrak{M}$ such that for all $\sigma \in \mathbb{T}$ and $\mathbf{C} \in \mathbb{C}^*_\sigma$

$$(\mathbf{C}) = [\mathbf{ind}_{e_0}(\mathbf{C})]^\sigma.\ \Box$$

4.4.16. Lemma. *For all $\sigma \in \mathbb{T}$ and $P \in \mathbf{CL}_\sigma(\mathbb{C}^*)^0$*

$$(P) \approx [\mathbf{ind}_{e_0}(P)]^\sigma.$$

Proof. By induction on the structure of P.

(i) For constants the result is trivial.

(ii) Let $\sigma,\tau \in \mathbb{T}$, $P \in \mathbf{CL}_{\sigma\to\tau}(\mathbb{C}^*)^0$ and $Q \in \mathbf{CL}_\sigma(\mathbb{C}^*)^0$. Then by induction and quasi-extensionality we have

$$\begin{aligned}(PQ) &= (P)(Q)\\ &\approx [\mathbf{ind}_{e_0}(P)]^{\sigma\to\tau}[\mathbf{ind}_{e_0}(Q)]^\sigma\\ &\approx [\mathbf{app}(\mathbf{ind}_{e_0}(P), \mathbf{ind}_{e_0}(Q))]^\tau\\ &= [\mathbf{ind}_{e_0}(PQ)]^\tau.\ \Box\end{aligned}$$

4.4.17. Lemma. (i) $(\mathsf{K}_{\sigma,\tau}) \approx \mathbf{k}_{\sigma,\tau}$.

(ii) $(\mathsf{S}_{\sigma,\tau,\upsilon}) \approx \mathbf{s}_{\sigma,\tau,\upsilon}$.

(iii) $(\mathbf{c}_m) = m$.

(iv) $(\mathbf{S}^+) \approx [s^+]^{0\to 0}$.

(v) $(\mathbf{\Delta}) \approx [\delta]^{t(4)}$.

Proof. By 4.4.4, the definition of **ind** and the definition of ( ). $\Box$

4.4.18. Proposition. *Let $m, n \in \omega$.*

(i) $(\mathbf{S}^+\mathbf{c}_m) = (\mathbf{c}_{m+1})$.

(ii) *For all* $a, b \in \mathfrak{M}_0$

(1) $(\mathbf{\Delta c}_m \mathbf{c}_n)ab = a$ *if* $m = n$

(2) $(\mathbf{\Delta c}_m \mathbf{c}_n)ab = b$ *if* $m \neq n$.

Proof. By (iii),(iv) and (v) of the previous lemma. □

The next proposition states the quasi-satisfaction of the 'combinatory' ε-axioms.

4.4.19. Proposition. *For all* $\sigma \in \mathbb{T}$ *and* $M \in \mathcal{T}_\sigma(\lambda\mathsf{E}(\mathcal{N}))^0$

$$(\mathbf{E}_\sigma \mathbf{c}_{\#M}) \approx (M_{CL}).$$

Proof. Let $\sigma \in \mathbb{T}$ and $M \in \mathcal{T}_\sigma(\lambda\mathsf{E}(\mathcal{N}))^0$. Then for all $\vec{a} \in (\mathfrak{M})_\sigma$

$$\begin{aligned}
(\mathbf{E}_\sigma \mathbf{c}_{\#M})\vec{a} &= (\mathbf{E}_\sigma)(\#M)\vec{a} \\
&= (\mathbf{E}_\sigma) \cdot \#M \cdot \vec{a} \\
&= [\mathbf{ind}_{e_0}(\mathbf{E}_\sigma)]^{0\to\sigma} \cdot \#M \cdot \vec{a} \\
&= \{\mathbf{ind}_{e_0}(\mathbf{E}_\sigma)\}^{0\to\sigma}(\#M, \vec{a}) \\
&= \{g(e_0, \#\sigma)\}^{0\to\sigma}(\#M, \vec{a}) \\
&= \mathcal{E}_\sigma(\mathbf{ind}_{e_0}(M_{CL}), \vec{a}) \\
&= \{\mathbf{ind}_{e_0}(M_{CL})\}^{\sigma}(\vec{a}) \\
&= [\mathbf{ind}_{e_0}(M_{CL})]^{\sigma}\vec{a} \\
&= (M_{CL})\vec{a}.
\end{aligned}$$

Therefore $(\mathbf{E}_\sigma \mathbf{c}_{\#M}) \approx (M_{CL})$. □

4.4.20. Definition. Let $[\![\,]\!] : \mathcal{T}(\lambda\mathsf{E}(\mathcal{N})) \times Val(\mathfrak{M}) \to |\mathfrak{M}|$ be defined by

$$[\![M]\!]_\rho = (M_{CL})_\rho. \ \square$$

The map $[\![\,]\!]$ is a quasi-$\lambda\mathsf{E}(\mathcal{N})$-interpretation in $\mathfrak{M}$, and so $(\mathfrak{M}, [\![\,]\!])$ is a quasi-$\lambda\mathsf{E}(\mathcal{N})$-structure.

4.4.21. Theorem. (i) $(\mathfrak{M}, [\![\,]\!])$ *is a quasi-*$\lambda\mathsf{E}\eta(\mathcal{N})$*-model with* $[\![\mathbf{c}_m]\!] = m$ *for all* $m \in \omega$.

(ii) *Every definable object of* $(\mathfrak{M}, [\![\,]\!])$ *is* $\mathcal{R}$*-recursive.*

PROOF. (i) First note that for all $m \in \omega$

$$[\![\mathbf{c}_m]\!] = (CL(\mathbf{c}_m)) = (\mathbf{c}_m) = m.$$

Because $\mathfrak{M}$ is quasi-extensional we only need to prove the quasi-satisfaction of the α-axioms and of the ε-axioms. The quasi-satisfaction of the α-axioms follows from 4.4.18. The quasi-satisfaction of the ε-axioms follows by 4.4.19 as follows. Let $\sigma \in \mathbb{T}$ and $M \in \mathcal{T}_\sigma(\lambda\mathsf{E}(\mathcal{N}))^0$. Then

$$\begin{aligned}[\![\mathbf{E}_\sigma \mathbf{c}_{\#M}]\!] &= (CL(\mathbf{E}_\sigma \mathbf{c}_{\#M})) \\ &= (\mathbf{E}_\sigma \mathbf{c}_{\#M}) \\ &\approx (M_{CL}) \\ &= [\![M]\!].\end{aligned}$$

So $\mathfrak{M}, [\![\ ]\!] \models_q \mathbf{E}_\sigma \mathbf{c}_{\#M} = M$. Therefore $(\mathfrak{M}, [\![\ ]\!])$ is a quasi-$\lambda\mathsf{E}\eta(\mathcal{N})$-model.

(ii) For all $\sigma \in \mathbb{T}$ and $\mathbf{C} \in \mathbb{C}_\sigma$

$$[\![\mathbf{C}]\!] = (CL(\mathbf{C})) = (\mathbf{C}) = [\mathbf{ind}_{e_0}(\mathbf{C})]^\sigma$$

so $[\![\mathbf{C}]\!]$ is an $\mathcal{R}$-recursive object. Therefore we have that for every $\sigma \in \mathbb{T}$

$$\mathfrak{D}_\sigma(\mathfrak{M}, [\![\ ]\!]) \subseteq \mathbf{Rec}_\sigma(\mathcal{R}). \ \Box$$

4.4.22. DEFINITION. (i) An ω-$\lambda\mathsf{E}\eta(\mathcal{N})$-model is a quasi-$\lambda\mathsf{E}\eta(\mathcal{N})$-model $(\mathfrak{M}, [\![\ ]\!])$ such that $\mathfrak{M}$ is an ω-structure and $[\![\mathbf{c}_m]\!] = m$ for all $m \in \omega$.

(ii) An ω-structure $\mathfrak{M}$ is called a pre-ω-$\lambda\mathsf{E}\eta(\mathcal{N})$-model if there exists a quasi-$\lambda\mathsf{E}(\mathcal{N})$-interpretation $[\![\ ]\!]$ in $\mathfrak{M}$ such that $(\mathfrak{M}, [\![\ ]\!])$ is an ω-$\lambda\mathsf{E}\eta(\mathcal{N})$-model. $\Box$

Collecting our results so far we obtain the following theorem.

4.4.23. THEOREM. *Let $\mathfrak{M}$ be an ω-structure and let $\mathcal{R}$ be a recursion theory on $\mathfrak{M}$. Then there exists a quasi-$\lambda\mathsf{E}(\mathcal{N})$-interpretation $[\![\ ]\!]$ in $\mathfrak{M}$ such that $(\mathfrak{M}, [\![\ ]\!])$ is an ω-$\lambda\mathsf{E}\eta(\mathcal{N})$-model and every definable object of $(\mathfrak{M}, [\![\ ]\!])$ is $\mathcal{R}$-recursive.* $\Box$

4.4.24. COROLLARY. *Let $\mathfrak{M}$ be an ω-structure. If $\mathfrak{M}$ is special, then $\mathfrak{M}$ is a pre-ω-$\lambda\mathsf{E}\eta(\mathcal{N})$-model.* $\Box$

In the following we will prove the inverse of this corollary.

Let $(\mathfrak{M}, [\![\ ]\!])$ be an ω-$\lambda\mathsf{E}\eta(\mathcal{N})$-model. We are going to construct a recursion theory $\mathcal{R}$ on $\mathfrak{M}$ with the property that the $\mathcal{R}$-recursive objects in $\mathfrak{M}$ are precisely the definable objects of $(\mathfrak{M}, [\![\ ]\!])$.

4.4.25. Definition. (i) For all $\sigma \in \mathbb{T}$ and $e \in \omega$ define $\{e\}^\sigma : (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ by

$$\{e\}^\sigma(\vec{a}) = [\![\mathbf{E}_\sigma \mathbf{c}_e]\!]\vec{a}.$$

(ii) Let $\mathcal{R} : \omega \times \mathbb{T} \to \mathbb{F}(\mathfrak{M})$ be defined by

$$\mathcal{R}(e, \sigma) = \{e\}^\sigma.$$

(iii) The set of indices is defined by

$$\mathbf{Ix} = \{\#M \mid M \in \mathcal{T}(\lambda\mathsf{E}(\mathcal{N}))^0\,\}.$$

(iv) Let $\mathbf{type} : \omega \to \omega$ be a recursive function such that for all $\sigma \in \mathbb{T}$ and all $M \in \mathcal{T}_\sigma(\lambda\mathsf{E}(\mathcal{N}))^0$

$$\mathbf{type}(\#M) = \#\sigma. \ \square$$

4.4.26. Proposition. *Let $\sigma \in \mathbb{T}$.*
(i) $\mathbf{Ix}(\sigma) = \{\#M \mid M \in \mathcal{T}_\sigma(\lambda\mathsf{E}(\mathcal{N}))^0\,\}$.
(ii) $\{\#M\}^\sigma = fnl([\![M]\!])$ *for every* $M \in \mathcal{T}_\sigma(\lambda\mathsf{E}(\mathcal{N}))^0$.
(iii) $\mathbf{REC}_\sigma(\mathcal{R}) = \{fnl([\![M]\!]) \mid M \in \mathcal{T}_\sigma(\lambda\mathsf{E}(\mathcal{N}))^0\,\}$.
(iv) $\mathbf{Rec}_\sigma(\mathcal{R}) = \mathfrak{D}_\sigma(\mathfrak{M}, [\![\ ]\!])$.

Proof. (i) Simple.
(ii) Let $M \in \mathcal{T}_\sigma(\lambda\mathsf{E}(\mathcal{N}))^0$. Then for all $\vec{a} \in (\mathfrak{M})_\sigma$

$$\begin{aligned} \{\#M\}^\sigma(\vec{a}) &= [\![\mathbf{E}_\sigma \mathbf{c}_{\#M}]\!]\vec{a} \\ &= [\![M]\!]\vec{a} \\ &= fnl([\![M]\!])(\vec{a}). \end{aligned}$$

(iii) We only need to prove $(\subseteq)$ because $(\supseteq)$ follows immediately from (ii). So let $\varphi \in \mathbf{REC}_\sigma(\mathcal{R})$. Then there exists $e \in \omega$ with $\varphi = \{e\}^\sigma$. Then for all $\vec{a} \in (\mathfrak{M})_\sigma$

$$\begin{aligned} \varphi(\vec{a}) &= \{e\}^\sigma(\vec{a}) \\ &= [\![\mathbf{E}_\sigma \mathbf{c}_e]\!]\vec{a} \\ &= fnl([\![\mathbf{E}_\sigma \mathbf{c}_e]\!])(\vec{a}). \end{aligned}$$

So $\varphi = fnl([\![\mathbf{E}_\sigma \mathbf{c}_e]\!])$.
(iv) By (iii). $\square$

4.4.27. Lemma. (i) C_0^0 *is $\mathcal{R}$-recursive.*
(ii) *There exists an $\mathcal{R}$-recursive* $\mathsf{S}^+ : \mathfrak{M}_0 \to \mathfrak{M}_0$ *such that* $\mathsf{S}^+(m) = m + 1$ *for all* $m \in \omega$.

(iii) *There exists an $\mathcal{R}$-recursive $\Delta : (\mathfrak{M}_0)^4 \to \mathfrak{M}_0$ such that for all $m, n \in \omega$ and $a, b \in \mathfrak{M}_0$*

(1) $\Delta(m, n, a, b) = a$ *if* $m = n$

(2) $\Delta(m, n, a, b) = b$ *if* $m \neq n$.

Proof. (i) $\mathsf{C}^0_0 = fnl(0) = fnl([\![\mathbf{c}_0]\!])$.
(ii) Take $\mathsf{S}^+ = fnl([\![\mathbf{S}^+]\!])$.
(iii) Take $\Delta = fnl([\![\mathbf{\Delta}]\!])$. □

4.4.28. Lemma. *$\mathcal{R}$ has the Application Property.*

Proof. Let $\mathbf{ap} : \omega \to \omega$ be a recursive function such that $\mathbf{ap}(\#\sigma) = \#(\mathbf{I}_\sigma)$ for all $\sigma \in \mathbb{T}$. Then $\mathbf{ap}(\#\sigma) \in \mathbf{Ix}(\sigma \to \sigma)$ for every $\sigma \in \mathbb{T}$ and for all $a \in \mathfrak{M}_\sigma$ and $\vec{b} \in (\mathfrak{M})_\sigma$

$$\begin{array}{rcl} \mathsf{Ap}_\sigma(a, \vec{b}) & = & a\vec{b} \\ & = & [\![\mathbf{I}_\sigma]\!]a\vec{b} \\ & = & fnl([\![\mathbf{I}_\sigma]\!])(a, \vec{b}) \\ & = & \{\#(\mathbf{I}_\sigma)\}^{\sigma\to\sigma}(a, \vec{b}) \\ & = & \{\mathbf{ap}(\#\sigma)\}^{\sigma\to\sigma}(a, \vec{b}). \end{array}$$

So $\mathsf{Ap}_\sigma = \{\mathbf{ap}(\#\sigma)\}^{\sigma\to\sigma}$. □

4.4.29. Lemma. *$\mathcal{R}$ has the Substitution Property.*

Proof. Let $\mathbf{sub} : \omega^3 \to \omega$ be a recursive function such that for all $\sigma, \tau \in \mathbb{T}$ with $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0$ and $\tau = \tau_1 \to \ldots \to \tau_l \to 0$, all $M \in \mathcal{T}_{\sigma*\tau}(\lambda\mathsf{E}(\mathcal{N}))^0$ and all $N_j \in \mathcal{T}_{\sigma*\tau_j}(\lambda\mathsf{E}(\mathcal{N}))^0$ $(j = 1, \ldots, l)$

$$\mathbf{sub}(\#\sigma, \#\tau, <\#M, \#N_1, \ldots, \#N_l>) = \#(\lambda\vec{x}.M\vec{x}(N_1\vec{x})\ldots(N_l\vec{x}))$$

with $\vec{x} = x_1, \ldots, x_l$ and $x_i \in Var_{\sigma_i}$ $(i = 1, \ldots, k)$.

Now let $\sigma, \tau \in \mathbb{T}$, $\tau = \tau_1 \to \ldots \to \tau_l \to 0$, let $e_0 \in \mathbf{Ix}(\sigma * \tau)$, $e_j \in \mathbf{Ix}(\sigma * \tau_j)$ $(j = 1, \ldots, l)$. Then there exists $M \in \mathcal{T}_{\sigma*\tau}(\lambda\mathsf{E}(\mathcal{N}))^0$ with $e_0 = \#M$, and for every $j \in \{1, \ldots, l\}$ there exists $N_j \in \mathcal{T}_{\sigma*\tau_j}(\lambda\mathsf{E}(\mathcal{N}))^0$ with $e_j = \#(N_j)$. Define

$$L \equiv \lambda\vec{x}.M\vec{x}(N_1\vec{x})\ldots(N_l\vec{x}) \in \mathcal{T}_\sigma(\lambda\mathsf{E}(\mathcal{N}))^0$$

$$\chi = \mathsf{Sub}_{\sigma,\tau}[\{e_0\}^{\sigma*\tau}, \{e_1\}^{\sigma*\tau_1}, \ldots, \{e_l\}^{\sigma*\tau_l}].$$

Then $\mathbf{sub}(\#\sigma, \#\tau, < e_0, e_1, \ldots, e_l >) = \#L \in \mathbf{Ix}(\sigma)$. We are going to show that $\chi = \{\#L\}^\sigma$. Let $\vec{a} \in (\mathfrak{M})_\sigma$. Define $b_j = [\![N_j]\!]\vec{a}$ $(j = 1, \ldots, l)$, and define $\vec{b} = (b_1, \ldots, b_l) \in (\mathfrak{M})_\tau$. Then for every $j \in \{1, \ldots, l\}$ and all $\vec{c} \in (\mathfrak{M})_{\tau_j}$

$$\begin{aligned} b_j\vec{c} &= [\![N_j]\!]\vec{a}\vec{c} \\ &= \{\#N_j\}^{\sigma*\tau_j}(\vec{a},\vec{c}) \\ &= \{e_j\}^{\sigma*\tau_j}(\vec{a},\vec{c}). \end{aligned}$$

Therefore we have

$$\begin{aligned} \chi(\vec{a}) &= \{e_0\}^{\sigma*\tau}(\vec{a},\vec{b}) \\ &= \{\#M\}^{\sigma*\tau}(\vec{a},\vec{b}) \\ &= [\![M]\!]\vec{a}\vec{b} \\ &= [\![M]\!]\vec{a}([\![N_1]\!]\vec{a})\ldots([\![N_l]\!]\vec{a}) \\ &= [\![M\vec{x}(N_1\vec{x})\ldots(N_l\vec{x})]\!]_{\rho(\vec{x}:=\vec{a})} \\ &= [\![L]\!]\vec{a} \\ &= \{\#L\}^{\sigma}(\vec{a}). \end{aligned}$$

So $\chi = \{\#L\}^\sigma = \{\mathbf{sub}(\#\sigma,\#\tau,< e_0,\ldots,e_l >)\}^\sigma$. □

4.4.30. Lemma. *$\mathcal{R}$ has the Manipulation Property.*

Proof. Let $\mathbf{man} : \omega^3 \to \omega$ be a recursive function such that for all $\sigma \in \mathbb{T}$ with $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0$, all $\vec{s} = (s_1,\ldots,s_l)$ with $1 \leq s_1,\ldots,s_l \leq k$, and all $M \in \mathcal{T}_{\sigma(\vec{s})}(\lambda\mathsf{E}(\mathcal{N}))^0$

$$\mathbf{man}(\#\sigma,< \vec{s} >,\#M) = \#(\lambda\vec{x}.Mx_{s_1}\ldots x_{s_l})$$

with $\vec{x} = x_1,\ldots,x_k$ and $x_i \in Var_{\sigma_i}$ $(i = 1,\ldots,k)$.

One can easily see that for all $\sigma \in \mathbb{T}$ with $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0$, all $\vec{s} = (s_1,\ldots,s_l)$ with $1 \leq s_1,\ldots,s_l \leq k$, and all $e \in \mathbf{Ix}(\sigma(\vec{s}))$ we have

(i) $\mathbf{man}(\#\sigma,< \vec{s} >,e) \in \mathbf{Ix}(\sigma)$

(ii) $\{\mathbf{man}(\#\sigma,< \vec{s} >,e)\}^\sigma = \mathsf{Man}_{\sigma,\vec{s}}[\{e\}^{\sigma(\vec{s})}]$. □

4.4.31. Lemma. *$\mathcal{R}$ has the Enumeration Property.*

Proof. Let $\mathbf{enum} : \omega \to \omega$ be a recursive function such that for all $\sigma \in \mathbb{T}$ we have $\mathbf{enum}(\#\sigma) = \#(\mathbf{E}_\sigma)$. Let $\sigma \in \mathbb{T}$. Then $\mathbf{enum}(\#\sigma) \in \mathbf{Ix}(0 \to \sigma)$. Now let $e \in \mathbf{Ix}(\sigma)$ and $\vec{a} \in (\mathfrak{M})_\sigma$. Then there exists $M \in \mathcal{T}_\sigma(\lambda\mathsf{E}(\mathcal{N}))^0$ with $e = \#M$. Then

$$\begin{aligned}
\{\mathbf{enum}(\#\sigma)\}^{0\to\sigma}(e,\vec{a}) &= \{\#(\mathbf{E}_\sigma)\}(e,\vec{a}) \\
&= [\![\mathbf{E}_\sigma]\!]e\vec{a} \\
&= [\![\mathbf{E}_\sigma]\!][\![\mathbf{c}_e]\!]\vec{a} \\
&= [\![\mathbf{E}_\sigma\mathbf{c}_e]\!]\vec{a} \\
&= [\![\mathbf{E}_\sigma\mathbf{c}_{\#M}]\!]\vec{a} \\
&= [\![M]\!]\vec{a} \\
&= \{\#M\}^\sigma(\vec{a}) \\
&= \{e\}^\sigma(\vec{a}).\ \Box
\end{aligned}$$

4.4.32. Theorem. *$\mathcal{R}$ is a recursion theory on $\mathfrak{M}$.*

Proof. By 4.4.27, 4.4.28, 4.4.29, 4.4.30 and 4.4.31. $\Box$

4.4.33. Theorem. *Let $(\mathfrak{M}, [\![\,]\!])$ be an ω-$\lambda\mathsf{E}\eta(\mathcal{N})$-model. Then there exists a recursion theory $\mathcal{R}$ on $\mathfrak{M}$ such that the $\mathcal{R}$-recursive objects in $\mathfrak{M}$ are precisely the definable objects of $(\mathfrak{M}, [\![\,]\!])$.* $\Box$

We see that every pre-ω-$\lambda\mathsf{E}\eta(\mathcal{N})$-model is special. Together with corollary 4.4.24 this gives us the following characterization of the special ω-structures.

4.4.34. Theorem. *Let $\mathfrak{M}$ be an ω-structure. Then $\mathfrak{M}$ is special if and only if $\mathfrak{M}$ is a pre-ω-$\lambda\mathsf{E}\eta(\mathcal{N})$-model.* $\Box$

4.5. A recursion theory on MON.

In this section we are going to construct a recursion theory on the type structure **MON**, thus showing that **MON** is a special ω-structure and a pre-ω-$\lambda\mathsf{E}\eta(\mathcal{N})$-model.

In the following $\mathfrak{M} = \mathbf{MON}$. Then $\mathfrak{M}$ is an extensional ω-structure. For every $\sigma \in \mathbb{T}$ we have an ordering $\sqsubseteq_\sigma$ on $\mathfrak{M}_\sigma$ such that $(\mathfrak{M}_\sigma, \sqsubseteq_\sigma)$ is a cpo, and for all $a, b \in \mathfrak{M}_\sigma$

$$a \sqsubseteq_\sigma b \Longleftrightarrow \forall \vec{c} \in (\mathfrak{M})_\sigma[\, a\vec{c} \sqsubseteq b\vec{c}\,]$$

where $\sqsubseteq$ is the ordering on $\mathfrak{M}_0 = \omega_\perp$ defined by

$$a \sqsubseteq b \Longleftrightarrow a = \perp \ \vee\ a = b.$$

Instead of $a \sqsubseteq_\sigma b$ we will usually write $a \sqsubseteq b$. For all $\sigma, \tau \in \mathbb{T}$ we have that $\mathfrak{M}_{\sigma\to\tau}$ is the set of all monotone functions from $\mathfrak{M}_\sigma$ into $\mathfrak{M}_\tau$. The $\mathfrak{M}$-representable functionals are precisely the monotone functionals on $\mathfrak{M}$.

We will also need the following lemma on $\mathfrak{M}$.

4.5.1. LEMMA. *Let $\sigma \in \mathbb{T}$, $a, a' \in \mathfrak{M}_\sigma$ be such that for all $\vec{b} \in (\mathfrak{M})_\sigma$*

$$a\vec{b} \in \omega \ \&\ a'\vec{b} \in \omega \Rightarrow a\vec{b} = a'\vec{b}.$$

Then there exists $a^* \in \mathfrak{M}_\sigma$ such that
(i) $a \sqsubseteq a^* \ \&\ a' \sqsubseteq a^*$
(ii) $\forall \vec{b} \in (\mathfrak{M})_\sigma [\, a^*\vec{b} \in \omega \Rightarrow a^*\vec{b} = a\vec{b} \ \vee\ a^*\vec{b} = a'\vec{b}\,]$.

PROOF. Define $\varphi : (\mathfrak{M})_\sigma \to \omega_\perp$ by
(i) $\varphi(\vec{b}) = a\vec{b}$ if $a\vec{b} \in \omega$
(ii) $\varphi(\vec{b}) = a'\vec{b}$ if $a\vec{b} = \perp$.
Then φ is monotone and therefore φ is $\mathfrak{M}$-representable. Let $a^* \in \mathfrak{M}_\sigma$ $\mathfrak{M}$-represent φ. This a^* has the required properties. Note that $a^* = sup\{a, a'\}$. □

In the following we are going to construct a recursion theory $\mathcal{R}$ on $\mathfrak{M}$. This will be done as follows. First we construct, by transfinite induction, a set $\Theta \subseteq \omega \times |\mathfrak{M}|^* \times \omega$. This Θ is something like a computation theory in general recursion theory ([Moschovakis 1969], [Fenstad 1980]). Then we define partial functionals φ_e^σ from (a subset of) $(\mathfrak{M})_\sigma$ into ω by setting

$$\varphi_e^\sigma(\vec{a}) \simeq m \Longleftrightarrow (e, \vec{a}, m) \in \Theta$$

for all $\sigma \in \mathbb{T}$, $e \in \omega$, $\vec{a} \in (\mathfrak{M})_\sigma$ and $m \in \omega$. Then we 'totalize' these partial functionals by letting the outcome be $\perp$ in the undefined points. So we define $\{e\}^\sigma : (\mathfrak{M})_\sigma \to \omega_\perp$ by

- $\{e\}^\sigma(\vec{a}) = \varphi_e^\sigma(\vec{a})$ if $\varphi_e^\sigma(\vec{a})\downarrow$
- $\{e\}^\sigma(\vec{a}) = \perp$ if $\varphi_e^\sigma(\vec{a})\uparrow$.

These functionals $\{e\}^\sigma$ then constitute a recursion theory $\mathcal{R}$ on $\mathfrak{M}$.

4.5.2. DEFINITION. (i) Let $\mathsf{S}^+ : \omega_\perp \to \omega_\perp$ be defined by
(1) $\mathsf{S}^+(m) = m + 1$ if $m \in \omega$
(2) $\mathsf{S}^+(m) = \perp$ if $m = \perp$.
(ii) Let $\Delta : (\omega_\perp)^4 \to \omega_\perp$ be defined by
(1) $\Delta(m, n, a, b) = a$ if $m \in \omega \ \&\ n \in \omega \ \&\ m = n$
(2) $\Delta(m, n, a, b) = b$ if $m \in \omega \ \&\ n \in \omega \ \&\ m \neq n$
(3) $\Delta(m, n, a, b) = \perp$ if $m \notin \omega \ \vee\ n \notin \omega$. □

4.5.3. LEMMA. (i) S^+ *is an $\mathfrak{M}$-representable functional of type* $0 \to 0$.
(ii) Δ *is an $\mathfrak{M}$-representable functional of type* $t(4)$.

PROOF. The functionals S^+ and Δ are monotone. $\Box$

4.5.4. DEFINITION. $\mathfrak{X} = \omega \times |\mathfrak{M}|^* \times \omega$. $\Box$

The set $\Theta \subseteq \mathfrak{X}$ is going to be defined inductively, by a monotone operator $\Phi : \wp(\mathfrak{X}) \to \wp(\mathfrak{X})$. But first we will give an 'informal' inductive definition of Θ.

4.5.5. DEFINITION. The set $\Theta \subseteq \mathfrak{X}$ is defined inductively as follows.

(i) $(< 0, \#0 >, (), 0) \in \Theta$.

(ii) For all $m \in \omega$

$$(< 1, \#(0 \to 0) >, (m), m+1) \in \Theta.$$

(iii) For all $m, n, a, b \in \mathfrak{M}_0$ with $\Delta(m, n, a, b) \in \omega$

$$(< 2, \#t(4) >, (m, n, a, b), \Delta(m, n, a, b)) \in \Theta.$$

(iv) (The Application Property.) For all $\sigma \in \mathbb{T}$, $a \in \mathfrak{M}_\sigma$, $\vec{b} \in (\mathfrak{M})_\sigma$ with $a\vec{b} \in \omega$

$$(< 3, \#(\sigma \to \sigma) >, (a, \vec{b}), a\vec{b}) \in \Theta.$$

(v) (The Substitution Property.) For all $\sigma, \tau \in \mathbb{T}$ with $\tau = \tau_1 \to \ldots \to \tau_l \to 0$, all $e_0, e_1, \ldots, e_l \in \omega$, $\vec{a} \in (\mathfrak{M})_\sigma$, $\vec{b} \in (\mathfrak{M})_\tau$, $m \in \omega$, if

(1) $(e_0, (\vec{a}, \vec{b}), m) \in \Theta$, and

(2) $\forall j \in \{1, \ldots, l\} \forall \vec{c} \in (\mathfrak{M})_{\tau_j} [\, b_j\vec{c} \in \omega \Rightarrow (e_j, (\vec{a}, \vec{c}), b_j\vec{c}) \in \Theta \,]$.

then

$$(< 4, \#\sigma, < e_0, e_1, \ldots, e_l >>, \vec{a}, m) \in \Theta.$$

(vi) (The Manipulation Property.) For all $\sigma \in \mathbb{T}$ with $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0$, all $\vec{s} = (s_1, \ldots, s_l)$ with $1 \leq s_1, \ldots, s_l \leq k$, all $e \in \omega$, all $\vec{a} \in (\mathfrak{M})_\sigma$ and all $m \in \omega$

$$(e, (\vec{a})_{\vec{s}}, m) \in \Theta \Rightarrow (< 5, \#\sigma, < e, \vec{s} >>, \vec{a}, m) \in \Theta.$$

(vii) (The Enumeration Property.) For all $\sigma \in \mathbb{T}$, $\vec{a} \in (\mathfrak{M})_\sigma$, $m \in \omega$

$$(e, \vec{a}, m) \in \Theta \Rightarrow (< 6, \#(0 \to \sigma) >, (e, \vec{a}), m) \in \Theta. \ \Box$$

4.5.6. DEFINITION. (i) Let $\mathbf{type} : \omega \to \omega$ be defined by

$$\mathbf{type}(e) = (e)_1.$$

(ii) A subset X of $\mathfrak{X}$ is consistent if for all $\sigma \in \mathbb{T}$, $e \in \omega$, $\vec{a}, \vec{b} \in (\mathfrak{M})_\sigma$ and $m \in \omega$

$$(e, \vec{a}, m) \in X \ \&\ \vec{a} \sqsubseteq \vec{b} \Rightarrow (e, \vec{b}, m) \in X.$$

(iii) A subset X of $\mathfrak{X}$ is single-valued if for all $\sigma \in \mathbb{T}$, $e \in \omega$, $\vec{a} \in (\mathfrak{M})_\sigma$ and $m, m' \in \omega$

$$(e, \vec{a}, m) \in X \;\&\; (e, \vec{a}, m') \in X \Rightarrow m = m'.$$

(iv) A subset X of $\mathfrak{X}$ is called a θ-set if X is consistent and single-valued and for all $\sigma \in \mathbb{T}$, $e \in \omega$, $\vec{a} \in (\mathfrak{M})_\sigma$ and $m \in \omega$

$$(e, \vec{a}, m) \in X \Rightarrow \mathbf{type}(e) = \#\sigma. \;\Box$$

It will be shown that Θ is a θ-set. But first we will define Θ by a formal inductive definition, using a monotone operator $\Phi : \wp(\mathfrak{X}) \to \wp(\mathfrak{X})$.

First we define a set $\mathsf{BAS} \in \wp(\mathfrak{X})$ of basic triples $(e, \vec{a}, m)$.

4.5.7. Definition. (i) $\mathsf{CON} = \{(<0, \#0>, (), 0)\}$.

(ii) SUC is the set of all triples

$$(<1, \#(0 \to 0)>, (m), m+1)$$

with $m \in \omega$.

(iii) EQ is the set of all triples

$$(<2, t(4)>, (m, n, a, b), \Delta(m, n, a, b))$$

with $m, n \in \omega$, $a, b \in \omega_\perp$ and $\Delta(m, n, a, b) \in \omega$.

(iv) AP is the set of all triples

$$(<3, \#(\sigma \to \sigma)>, (a, \vec{b}), a\vec{b})$$

with $\sigma \in \mathbb{T}$, $a \in \mathfrak{M}_\sigma$, $\vec{b} \in (\mathfrak{M})_\sigma$ and $a\vec{b} \in \omega$.

(v) $\mathsf{BAS} = \mathsf{CON} \cup \mathsf{SUC} \cup \mathsf{EQ} \cup \mathsf{AP}$. $\Box$

Next we will define three operators SUB, MAN and ENUM on $\mathfrak{X}$ that introduce substitution, manipulation and enumeration.

4.5.8. Definition. (i) Let $\mathsf{SUB} : \wp(\mathfrak{X}) \to \wp(\mathfrak{X})$ be defined as follows. Let $X \in \wp(\mathfrak{X})$. Then $\mathsf{SUB}(X)$ is the set of all triples

$$(<4, \#\sigma, <e_0, e_1, \ldots, e_l>>, \vec{a}, m)$$

with $\sigma \in \mathbb{T}$, $l \in \omega$, $e_0, e_1, \ldots, e_l \in \omega$, $\vec{a} \in (\mathfrak{M})_\sigma$ and $m \in \omega$ such that there exist $\tau = \tau_1 \to \ldots \to \tau_l \to 0 \in \mathbb{T}$ and $\vec{b} = (b_1, \ldots, b_l) \in (\mathfrak{M})_\tau$ with

(1) $\mathbf{type}(e_0) = \#(\sigma * \tau)$

(2) $\mathbf{type}(e_j) = \#(\sigma * \tau_j)$ $(j = 1, \ldots, l)$

(3) $(e_0, (\vec{a}, \vec{b}), m) \in X$

(4) $\forall j \in \{1, \ldots, l\} \forall \vec{c} \in (\mathfrak{M})_{\tau_j} [\, b_j\vec{c} \in \omega \Rightarrow (e_j, (\vec{a}, \vec{c}), b_j\vec{c}) \in X \,]$.

(ii) Let $\mathsf{MAN} : \wp(\mathfrak{X}) \to \wp(\mathfrak{X})$ be defined as follows. Let $X \in \wp(\mathfrak{X})$. Then $\mathsf{MAN}(X)$ is the set of all triples

$$(<5, \#\sigma, <e, \vec{s}>>, \vec{a}, m)$$

with $\sigma \in \mathbb{T}$, $e \in \omega$, $\vec{s} = (s_1, \ldots, s_l)$, $1 \leq s_1, \ldots, s_l \leq fac(\sigma)$, $\vec{a} \in (\mathfrak{M})_\sigma$, $m \in \omega$ and $(e, (\vec{a})_{\vec{s}}, m) \in X$.

(iii) Let $\mathsf{ENUM} : \wp(\mathfrak{X}) \to \wp(\mathfrak{X})$ be defined as follows. Let $X \in \wp(\mathfrak{X})$. Then $\mathsf{ENUM}(X)$ is the set of all triples

$$(<6, \#(0 \to \sigma)>, (e, \vec{a}), m)$$

with $\sigma \in \mathbb{T}$, $e \in \omega$, $\vec{a} \in (\mathfrak{M})_\sigma$ and $(e, \vec{a}, m) \in X$. □

4.5.9. Definition. Let $\Phi : \wp(\mathfrak{X}) \to \wp(\mathfrak{X})$ be defined by

$$\Phi(X) = \mathsf{BAS} \cup \mathsf{SUB}(X) \cup \mathsf{MAN}(X) \cup \mathsf{ENUM}(X). \ \square$$

The operator Φ is a monotone operator on $\mathfrak{X}$ because SUB, MAN and ENUM are monotone operators on $\mathfrak{X}$.

Recall that the stages Φ^ζ of Φ are defined by $\Phi^\zeta = \Phi(\Phi^{(\zeta)})$ with $\Phi^{(\zeta)} = \bigcup\{\Phi^\xi \mid \xi < \zeta\}$ for all $\zeta \in \mathbf{On}$. Also recall that the set which is defined inductively by Φ is $\Phi^\infty = \bigcup\{\Phi^\zeta \mid \zeta \in \mathbf{On}\}$.

4.5.10. Definition. $\Theta = \Phi^\infty$. □

First it will be shown that Θ is a θ-set.

4.5.11. Definition. Let $X \in \wp(\mathfrak{X})$. The domain of X is defined by

$$\mathbf{dom}(X) = \{(e, \vec{a}) \mid \exists m \in \omega[\,(e, \vec{a}, m) \in X\,]\}. \ \square$$

4.5.12. Lemma. *If $X_1, \ldots, X_n$ are θ-sets with mutually disjoint domains, then the union $X_1 \cup \ldots \cup X_n$ is also a θ-set.*

Proof. Simple. (The disjointness of the domains is used in showing single-valuedness.) □

4.5.13. Lemma. *BAS is a θ-set.*

Proof. It is easy to see that CON, SUC, EQ and AP are θ-sets. Their domains are disjoint, because

$$(e, \vec{a}, m) \in \mathsf{CON} \Rightarrow (e)_0 = 0$$
$$(e, \vec{a}, m) \in \mathsf{SUC} \Rightarrow (e)_0 = 1$$
$$(e, \vec{a}, m) \in \mathsf{EQ} \Rightarrow (e)_0 = 2$$
$$(e, \vec{a}, m) \in \mathsf{AP} \Rightarrow (e)_0 = 3.$$

So by the previous lemma BAS is a θ-set. □

4.5.14. LEMMA. *If X is a θ-set, then* $\mathsf{SUB}(X)$, $\mathsf{MAN}(X)$ *and* $\mathsf{ENUM}(X)$ *are also θ-sets.*

PROOF. Let X be a θ-set.

(i) We are going to show that $\mathsf{SUB}(X)$ is a θ-set. First we will show consistency. Let $(e, \vec{a}, m) \in \mathsf{SUB}(X)$ and let $\vec{a} \sqsubseteq \vec{a}'$. Then $e =< 4, \#\sigma, < e_0, e_1, \ldots, e_l >>$ for some $\sigma \in \mathbb{T}$, $l \in \omega$ and $e_0, e_1, \ldots, e_l \in \omega$ with $\vec{a}, \vec{a}' \in (\mathfrak{M})_\sigma$ and such that there exist $\tau \in \mathbb{T}$ with $\tau = \tau_1 \to \ldots \to \tau_l \to 0$, and $\vec{b} \in (\mathfrak{M})_\sigma$ with

(1) $\mathbf{type}(e_0) = \#(\sigma * \tau)$

(2) $\mathbf{type}(e_j) = \#(\sigma * \tau_j)$ $(j = 1, \ldots, l)$

(3) $(e_0, (\vec{a}, \vec{b}), m) \in X$

(4) $\forall j \in \{1, \ldots, l\} \forall \vec{c} \in (\mathfrak{M})_{\tau_j}[\, b_j\vec{c} \in \omega \Rightarrow (e_j, (\vec{a}, \vec{c}), b_j\vec{c}) \in X\,]$.

Because X is consistent, (3) gives $(e_0, (\vec{a}', \vec{b}), m) \in X$, and (4) gives, for all $j \in \{1, \ldots, l\}$, $\vec{c} \in (\mathfrak{M})_{\tau_j}$

$$b_j\vec{c} \in \omega \Rightarrow (e_j, (\vec{a}', \vec{c}), b_j\vec{c}) \in X.$$

Therefore $(e, \vec{a}', m) \in \mathsf{SUB}(X)$. So $\mathsf{SUB}(X)$ is consistent.

Next we show that $\mathsf{SUB}(X)$ is single-valued. Let $(e, \vec{a}, m) \in \mathsf{SUB}(X)$ and $(e, \vec{a}, m') \in \mathsf{SUB}(X)$. Then $e =< 4, \#\sigma, < e_0, e_1, \ldots, e_l >>$ for some $\sigma \in \mathbb{T}$, $l \in \omega$ and $e_0, e_1, \ldots, e_l \in \omega$ with $\vec{a} \in (\mathfrak{M})_\sigma$. Because $(e, \vec{a}, m) \in \mathsf{SUB}(X)$, there exist $\tau = \tau_1 \to \ldots \tau_l \to 0 \in \mathbb{T}$ and $\vec{b} \in (\mathfrak{M})_\tau$ with

(1) $\mathbf{type}(e_0) = \#(\sigma * \tau)$

(2) $\mathbf{type}(e_j) = \#(\sigma * \tau_j)$ $(j = 1, \ldots, l)$

(3) $(e_0, (\vec{a}, \vec{b}), m) \in X$

(4) $\forall j \in \{1, \ldots, l\} \forall \vec{c} \in (\mathfrak{M})_{\tau_j}[\, b_j\vec{c} \in \omega \Rightarrow (e_j, (\vec{a}, \vec{c}), b_j\vec{c}) \in X\,]$.

And by $(e, \vec{a}, m') \in \mathsf{SUB}(X)$ there exist $\tau' = \tau'_1 \to \ldots \to \tau'_l \to 0$ and $\vec{b}' \in (\mathfrak{M})_{\tau'}$ with

(1) $\mathbf{type}(e_0) = \#(\sigma * \tau')$

(2) $\mathbf{type}(e_j) = \#(\sigma * \tau'_j)$ $(j = 1, \ldots, l)$

(3) $(e_0, (\vec{a}, \vec{b}'), m') \in X$

(4) $\forall j \in \{1, \ldots, l\} \forall \vec{c} \in (\mathfrak{M})_{\tau'_j}[\, b'_j\vec{c} \in \omega \Rightarrow (e_j, (\vec{a}, \vec{c}), b'_j\vec{c}) \in X\,]$.

Then we have $\tau = \tau'$ and $\tau_j = \tau'_j$ $(j = 1, \ldots, l)$. Let $j \in \{1, \ldots, l\}$. Let $\vec{c} \in (\mathfrak{M})_{\tau_j}$ be such that $b_j\vec{c} \in \omega$ and $b'_j\vec{c} \in \omega$. Then

$$(e_j, (\vec{a}, \vec{c}), b_j\vec{c}) \in X$$

$$(e_j, (\vec{a}, \vec{c}), b'_j\vec{c}) \in X.$$

But X is single-valued, so $b_j\vec{c} = b'_j\vec{c}$. Therefore, for every $j \in \{1, \ldots, l\}$

$$\forall \vec{c} \in (\mathfrak{M})_{\tau_j}[\, b_j\vec{c} \in \omega \;\&\; b'_j\vec{c} \in \omega \Rightarrow b_j\vec{c} = b'_j\vec{c}\,].$$

Then by 4.5.1 there exist $b^*_j \in \mathfrak{M}_{\tau_j}$ $(j = 1, \ldots, l)$ such that $b_j \sqsubseteq b^*_j$ and $b'_j \sqsubseteq b^*_j$ and for all $\vec{c} \in (\mathfrak{M})_{\tau_j}$

$$b^*_j\vec{c} \in \omega \Rightarrow b^*_j\vec{c} = b_j\vec{c} \vee b^*_j\vec{c} = b'_j\vec{c}.$$

Define $\vec{b}^* = (b^*_1, \ldots, b^*_l) \in (\mathfrak{M})_\tau$. Then $\vec{b} \sqsubseteq \vec{b}^*$ and $\vec{b}' \sqsubseteq \vec{b}^*$, and therefore, by the consistency of X it follows that

$$(e_0, (\vec{a}, \vec{b}^*), m) \in X$$

$$(e_0, (\vec{a}, \vec{b}^*), m') \in X.$$

Again by the single-valuedness of X we get $m = m'$. Therefore $\mathsf{SUB}(X)$ is single-valued.

Let $(e, \vec{a}, m) \in \mathsf{SUB}(X)$ with $\sigma \in \mathbb{T}$ and $\vec{a} \in (\mathfrak{M})_\sigma$. Then $e =< 4, \#\sigma, < e_0, \ldots, e_l >>$ for some $e_0, \ldots, e_l \in \omega$, and so $\mathbf{type}(e) = (e)_1 = \#\sigma$.

Altogether we have that $\mathsf{SUB}(X)$ is a θ-set.

(ii) It is easy to see that also $\mathsf{MAN}(X)$ and $\mathsf{ENUM}(X)$ are θ-sets. $\square$

4.5.15. Lemma. *If X is a θ-set, then $\Phi(X)$ is also a θ-set.*

Proof. Let X be a θ-set. Then $\mathsf{SUB}(X)$, $\mathsf{MAN}(X)$ and $\mathsf{ENUM}(X)$ are θ-sets. Also BAS is a θ-set. The domains of BAS, $\mathsf{SUB}(X)$, $\mathsf{MAN}(X)$ and $\mathsf{ENUM}(X)$ are disjoint, because

$$(e, \vec{a}, m) \in \mathsf{BAS} \Rightarrow (e)_0 \in \{0, 1, 2, 3\}$$

$$(e, \vec{a}, m) \in \mathsf{SUB}(X) \Rightarrow (e)_0 = 4$$

$$(e, \vec{a}, m) \in \mathsf{MAN}(X) \Rightarrow (e)_0 = 5$$

$$(e, \vec{a}, m) \in \mathsf{ENUM}(X) \Rightarrow (e)_0 = 6.$$

Therefore $\Phi(X)$ is a θ-set. $\square$

4.5.16. Lemma. (i) *If $(X_i)_{i\in I}$ is a chain of θ-sets (that is $X_i \subseteq X_j$ or $X_j \subseteq X_i$ for all $i, j \in I$), then $\bigcup\{X_i |\, i \in I\,\}$ is also a θ-set.*

(ii) *For every $\zeta \in \mathbf{On}$ we have that Φ^ζ is a θ-set.*

Proof. (i) Simple.

(ii) By transfinite induction on ζ, using (i) and the previous lemma. $\square$

4.5.17. Proposition. *Θ is a θ-set.*

PROOF. There exists $\zeta \in \mathbf{On}$ with $\Theta = \Phi^\infty = \Phi^\zeta$, so by (ii) of the previous lemma it follows that Θ is a θ-set. $\Box$

4.5.18. DEFINITION. For all $\sigma \in \mathbb{T}$ and $e \in \omega$ define

$$\varphi_e^\sigma = \{(\vec{a}, m) \in (\mathfrak{M})_\sigma \times \omega |\, (e, \vec{a}, m) \in \Theta\,\}. \;\Box$$

4.5.19. LEMMA. (i) *φ_e^σ is a partial map from $(\mathfrak{M})_\sigma$ into ω.*
(ii) $\varphi_e^\sigma(\vec{a})\downarrow \;\&\; \vec{a} \sqsubseteq \vec{b} \Rightarrow \varphi_e^\sigma(\vec{a}) \simeq \varphi_e^\sigma(\vec{b})$.

PROOF. (i) By the single-valuedness of Θ.
(ii) By the consistency of Θ. $\Box$

4.5.20. DEFINITION. For all $\sigma \in \mathbb{T}$ and $e \in \omega$ define $\{e\}^\sigma : (\mathfrak{M})_\sigma \to \omega_\perp$ by
(i) $\{e\}^\sigma(\vec{a}) = \varphi_e^\sigma(\vec{a})$ if $\varphi_e^\sigma(\vec{a})\downarrow$
(ii) $\{e\}^\sigma(\vec{a}) = \perp$ if $\varphi_e^\sigma(\vec{a})\uparrow$. $\Box$

4.5.21. LEMMA. *Let $\sigma \in \mathbb{T}$, $e \in \omega$.*
(i) *$\{e\}^\sigma$ is monotone.*
(ii) *$\{e\}^\sigma$ is $\mathfrak{M}$-representable.*

PROOF. (i) Let $\vec{a}, \vec{b} \in (\mathfrak{M})_\sigma$ with $\vec{a} \sqsubseteq \vec{b}$. Suppose $\{e\}^\sigma(\vec{a}) \in \omega$. Then $\varphi_e^\sigma(\vec{a})\downarrow$ and $\{e\}^\sigma(\vec{a}) = \varphi_e^\sigma(\vec{a})$. Then $\varphi_e^\sigma(\vec{a}) \simeq \varphi_e^\sigma(\vec{b})$, so $\varphi_e^\sigma(\vec{b})\downarrow$, so $\{e\}^\sigma(\vec{b}) = \varphi_e^\sigma(\vec{b})$. So $\{e\}^\sigma(\vec{a}) = \{e\}^\sigma(\vec{b})$. Therefore $\{e\}^\sigma(\vec{a}) \sqsubseteq \{e\}^\sigma(\vec{b})$.
(ii) By (i). $\Box$

4.5.22. LEMMA. *Let $e \in \omega$, $\sigma \in \mathbb{T}$, $\vec{a} \in (\mathfrak{M})_\sigma$, $m \in \omega$.*
(i) $(e, \vec{a}, m) \in \Theta \iff \{e\}^\sigma(\vec{a}) = m$.
(ii) $(e, \vec{a}) \notin \mathbf{dom}(\Theta) \iff \{e\}^\sigma(\vec{a}) = \perp$.

PROOF. Simple. $\Box$

From (ii) of the previous lemma it follows that

$$\mathbf{type}(e) \neq \#\sigma \Rightarrow \{e\}^\sigma(\vec{a}) = \perp.$$

4.5.23. DEFINITION. (i) Let $\mathcal{R} : \omega \times \mathbb{T} \to \mathbb{F}(\mathfrak{M})$ be defined by

$$\mathcal{R}(e, \sigma) = \{e\}^\sigma.$$

(ii) $\mathbf{Ix} = \{e \in \omega |\, \exists \sigma \in \mathbb{T}[\,\mathbf{type}(e) = \#\sigma\,]\,\}$. $\Box$

We are going to show that $\mathcal{R}$ is a recursion theory on $\mathfrak{M} = \mathbf{MON}$.

Note that $\mathbf{Ix}$ is a recursive subset of ω, and that for every $\sigma \in \mathbb{T}$

$$\mathbf{Ix}(\sigma) = \{e \in \omega |\, \mathbf{type}(e) = \#\sigma\,\}.$$

4.5.24. LEMMA. *For every $\sigma \in \mathbb{T}$ and all $\varphi \in \mathbf{REC}_\sigma(\mathcal{R})$ there exists $e \in \mathbf{Ix}(\sigma)$ with $\varphi = \{e\}^\sigma$.*

PROOF. Simple. □

In the following bear in mind the fact that Θ is a fixed point of Φ.

4.5.25. LEMMA. *Let $(e, \vec{a}, m) \in \Theta$.*
(i) $(e)_0 = 0 \Rightarrow (e, \vec{a}, m) \in \mathsf{CON}$.
(ii) $(e)_0 = 1 \Rightarrow (e, \vec{a}, m) \in \mathsf{SUC}$.
(iii) $(e)_0 = 2 \Rightarrow (e, \vec{a}, m) \in \mathsf{EQ}$.
(iv) $(e)_0 = 3 \Rightarrow (e, \vec{a}, m) \in \mathsf{AP}$.
(v) $(e)_0 = 4 \Rightarrow (e, \vec{a}, m) \in \mathsf{SUB}(\Theta)$.
(vi) $(e)_0 = 5 \Rightarrow (e, \vec{a}, m) \in \mathsf{MAN}(\Theta)$.
(vii) $(e)_0 = 6 \Rightarrow (e, \vec{a}, m) \in \mathsf{ENUM}(\Theta)$.

PROOF. By the fact that

$$\Theta = \mathsf{CON} \cup \mathsf{SUC} \cup \mathsf{EQ} \cup \mathsf{AP} \cup \mathsf{SUB}(\Theta) \cup \mathsf{MAN}(\Theta) \cup \mathsf{ENUM}(\Theta). \ \square$$

In the following we will often use the next lemma for proving the equality of two functionals.

4.5.26. LEMMA. *Let $\sigma \in \mathbb{T}$, $\varphi, \psi : (\mathfrak{M})_\sigma \to \omega_\perp$. Suppose that for all $\vec{a} \in (\mathfrak{M})_\sigma$ and $m \in \omega$*

$$\varphi(\vec{a}) = m \iff \psi(\vec{a}) = m.$$

Then $\varphi = \psi$.

PROOF. Let $\vec{a} \in (\mathfrak{M})_\sigma$. Then we have

$$\varphi(\vec{a}) \in \omega \Rightarrow \varphi(\vec{a}) = \psi(\vec{a})$$

and so $\varphi(\vec{a}) \sqsubseteq \psi(\vec{a})$. Likewise $\psi(\vec{a}) \sqsubseteq \varphi(\vec{a})$. So $\varphi(\vec{a}) = \psi(\vec{a})$. □

4.5.27. LEMMA. *C^0_0, S^+ and Δ are $\mathcal{R}$-recursive.*

PROOF. (i) Because $(< 0, \#0 >, (), 0) \in \mathsf{CON} \subseteq \Theta$ we have $\{< 0, \#0 >\}^0() = 0 = \mathsf{C}^0_0()$ so $\mathsf{C}^0_0 = \{< 0, \#0 >\}^0$.

(ii) Using 4.5.26 we are going to show that $\mathsf{S}^+ = \{< 1, \#(0\to 0) >\}^{0\to 0}$. Define $e =< 1, \#(0\to 0) >$. Let $m \in \mathfrak{M}_0$ and $n \in \omega$.

($\Rightarrow$) Suppose $\mathsf{S}^+(m) = n$. Then $m \in \omega$ and $n = m + 1$. So $(e, (m), n) \in \mathsf{SUC} \subseteq \Theta$, so $\{e\}^{0\to 0}(m) = n$.

($\Leftarrow$) Now suppose $\{e\}^{0\to 0}(m) = n$. Then $(e, (m), n) \in \Theta$. Because $(e)_0 = 1$ we have $(e, (m), n) \in \mathsf{SUC}$, so $n = m + 1$, and therefore $\mathsf{S}^+(m) = n$.

(iii) Using 4.5.26 it is easy to show $\Delta = \{< 2, \#t(4) >\}^{t(4)}$. □

4.5.28. PROPOSITION. *$\mathcal{R}$ has the Application Property.*

PROOF. Let $\mathbf{ap} : \omega \to \omega$ be a recursive function such that for all $\sigma \in \mathbb{T}$ we have $\mathbf{ap}(\#\sigma) =< 3, \#(\sigma \to \sigma) >$.

Let $\sigma \in \mathbb{T}$. Using 4.5.26 we are going to show

$$\mathsf{Ap}_\sigma = \{\mathbf{ap}(\#\sigma)\}^{\sigma\to\sigma}.$$

Define $e = \mathbf{ap}(\#\sigma)$. Let $(a, \vec{b}) \in \mathfrak{M}_\sigma \times (\mathfrak{M})_\sigma$, $m \in \omega$.

($\Rightarrow$) Suppose $\mathsf{Ap}_\sigma(a, \vec{b}) = m$. Then $a\vec{b} = m$, so $(e, (a, \vec{b}), m) \in \mathsf{AP} \subseteq \Theta$, so $\{e\}^{\sigma\to\sigma}(a, \vec{b}) = m$.

($\Leftarrow$) Now suppose $\{e\}^{\sigma\to\sigma}(a, \vec{b}) = m$. Then $(e, (a, \vec{b}), m) \in \Theta$. Because $(e)_0 = 3$ we have $(e, (a, \vec{b}), m) \in \mathsf{AP}$, so $m = a\vec{b}$, and so $\mathsf{Ap}_\sigma(a, \vec{b}) = m$. $\Box$

In order to show that $\mathcal{R}$ has the Substitution Property, we need to make a few preparations.

4.5.29. DEFINITION. For all $\sigma, \tau \in \mathbb{T}$, $e \in \mathbf{Ix}(\sigma * \tau)$ and $\vec{a} \in (\mathfrak{M})_\sigma$ define

$$\mathsf{Z}(e, \vec{a}) = \{b \in \mathfrak{M}_\tau \,|\, \forall \vec{c} \in (\mathfrak{M})_\tau[\, b\vec{c} \in \omega \Rightarrow (e, (\vec{a}, \vec{c}), b\vec{c}) \in \Theta\,]\}. \ \Box$$

4.5.30. LEMMA. *Let $\sigma, \tau \in \mathbb{T}$, $e \in \mathbf{Ix}(\sigma * \tau)$ and $\vec{a} \in (\mathfrak{M})_\sigma$. Let $z \in \mathfrak{M}_\tau$ be such that for all $\vec{c} \in (\mathfrak{M})_\tau$*

$$z\vec{c} = \{e\}^{\sigma*\tau}(\vec{a}, \vec{c}).$$

Then $z = \mathbf{sup}(\mathsf{Z}(e, \vec{a}))$.

PROOF. (i) Let $b \in \mathsf{Z}(e, \vec{a})$. Let $\vec{c} \in (\mathfrak{M})_\tau$, and suppose $b\vec{c} \in \omega$. Then we have that $(e, (\vec{a}, \vec{c}), b\vec{c}) \in \Theta$, by the definition of $\mathsf{Z}(e, \vec{a})$. So $b\vec{c} = \{e\}^{\sigma*\tau}(\vec{a}, \vec{c}) = z\vec{c}$. So

$$\forall \vec{c} \in (\mathfrak{M})_\tau[\, b\vec{c} \in \omega \Rightarrow b\vec{c} = z\vec{c}\,]$$

and so $b \sqsubseteq z$. Therefore $\mathsf{Z}(e, \vec{a}) \sqsubseteq z$.

(ii) For all $\vec{c} \in (\mathfrak{M})_\tau$

$$z\vec{c} \in \omega \Rightarrow (e, (\vec{a}, \vec{c}), z\vec{c}) \in \Theta$$

so $z \in \mathsf{Z}(e, \vec{a})$.

From (i) and (ii) it follows that $\mathbf{sup}(\mathsf{Z}(e, \vec{a})) = z$. $\Box$

4.5.31. PROPOSITION. *$\mathcal{R}$ has the Substitution Property.*

PROOF. Let $\mathbf{sub} : \omega^3 \to \omega$ be a recursive function such that for all $\sigma, \tau \in \mathbb{T}$ and all $e_0, e_1, \ldots, e_l \in \omega$

$$\mathbf{sub}(\#\sigma, \#\tau, < e_0, \ldots, e_l >) =< 4, \#\sigma, < e_0, \ldots, e_l >> .$$

Let $\sigma, \tau \in \mathbb{T}$, $\tau = \tau_1 \to \ldots \to \tau_l \to 0$, $e_0 \in \mathbf{Ix}(\sigma * \tau)$, $e_j \in \mathbf{Ix}(\sigma * \tau_j)$ $(j = 1, \ldots, l)$. Now define

$$e = \mathbf{sub}(\#\sigma, \#\tau, < e_0, e_1, \ldots, e_l >)$$
$$\chi = \mathsf{Sub}_{\sigma,\tau}[\{e_0\}^{\sigma*\tau}, \{e_1\}^{\sigma*\tau_1}, \ldots, \{e_l\}^{\sigma*\tau_l}].$$

Using 4.5.26 we are going to show that $\chi = \{e\}^\sigma$. Let $\vec{a} \in (\mathfrak{M})_\sigma$ and $m \in \omega$.

($\Rightarrow$) Suppose $\chi(\vec{a}) = m$. Then there exists $\vec{b} \in (\mathfrak{M})_\tau$ such that $\{e_0\}^{\sigma*\tau}(\vec{a}, \vec{b}) = m$ and for every $j \in \{1, \ldots, l\}$ and all $\vec{c} \in (\mathfrak{M})_{\tau_j}$

$$b_j\vec{c} = \{e_j\}^{\sigma*\tau_j}(\vec{a}, \vec{c}).$$

Then $(e_0, (\vec{a}, \vec{b}), m) \in \Theta$ and for all $j \in \{1, \ldots, l\}$ and all $\vec{c} \in (\mathfrak{M})_{\tau_j}$

$$b_j\vec{c} \in \omega \Rightarrow (e_j, (\vec{a}, \vec{c}), b_j\vec{c}) \in \Theta.$$

Therefore $(e, \vec{a}, m) \in \mathsf{SUB}(\Theta) \subseteq \Phi(\Theta) = \Theta$, and so $\{e\}^\sigma = m$.

($\Leftarrow$) Now suppose $\{e\}^\sigma(\vec{a}) = m$. Then $(e, \vec{a}, m) \in \Theta$. Because $(e)_0 = 4$ we have $(e, \vec{a}, m) \in \mathsf{SUB}(\Theta)$. So there exists $v \in \mathbb{T}$ with $v = v_1 \to \ldots \to v_l \to 0$ and there exists $\vec{b} \in (\mathfrak{M})_v$ such that

(i) $\mathbf{type}(e_0) = \#(\sigma * v)$
(ii) $\mathbf{type}(e_j) = \#(\sigma * v_j)$ $(j = 1, \ldots, l)$
(iii) $(e_0, (\vec{a}, \vec{b}), m) \in \Theta$
(iv) $\forall j \in \{1, \ldots, l\} \forall \vec{c} \in (\mathfrak{M})_{v_j} [\, b_j\vec{c} \in \omega \Rightarrow (e_j, (\vec{a}, \vec{c}), b_j\vec{c}) \in \Theta \,]$.

By (i) and $e_0 \in \mathbf{Ix}(\sigma * \tau)$ we have $\tau = v$ and $\tau_j = v_j$ $(j = 1, \ldots, l)$. For every $j \in \{1, \ldots, l\}$ let $z_j \in \mathfrak{M}_{\tau_j}$ be such that for all $\vec{c} \in (\mathfrak{M})_{\tau_j}$

$$z_j\vec{c} = \{e_j\}^{\sigma*\tau_j}(\vec{a}, \vec{c}).$$

Then $z_j = \mathbf{sup}(\mathsf{Z}(e_j, \vec{a}))$ for every $j \in \{1, \ldots, l\}$, by the previous lemma. Define $\vec{z} = (z_1, \ldots, z_l) \in (\mathfrak{M})_\tau$. By (iv) we have $b_j \in \mathsf{Z}(e_j, \vec{a})$ and so $b_j \sqsubseteq z_j$ for every $j \in \{1, \ldots, l\}$. So $\vec{b} \sqsubseteq \vec{z}$. Then by (iii) and the consistency of Θ it follows that $(e_0, (\vec{a}, \vec{z}), m) \in \Theta$, and so $\{e_0\}^{\sigma*\tau}(\vec{a}, \vec{z}) = m$. So $\chi(\vec{a}) = m$. □

4.5.32. PROPOSITION. *$\mathcal{R}$ has the Manipulation Property.*

PROOF. Let $\mathbf{man} : \omega^3 \to \omega$ be a recursive function such that for all $\sigma \in \mathbb{T}$, $\vec{s} = (s_1, \ldots, s_l)$, $e \in \omega$

$$\mathbf{man}(\#\sigma, < \vec{s} >, e) =< 5, \#\sigma, < e, \vec{s} >> .$$

Let $\sigma \in \mathbb{T}$, $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0$, $\vec{s} = (s_1, \ldots, s_l)$, $1 \leq s_1, \ldots, s_l \leq k$, $e \in \mathbf{Ix}(\sigma(\vec{s}))$. Using 4.5.26 it is easy to show that

$$\{\mathbf{man}(\#\sigma, < \vec{s} >, e)\}^\sigma = \mathsf{Man}_{\sigma,\vec{s}}[\{e\}^{\sigma(\vec{s})}]. \ \Box$$

4.5.33. PROPOSITION. *$\mathcal{R}$ has the Enumeration Property.*

PROOF. Let **enum** : $\omega \to \omega$ be a recursive function such that for all $\sigma \in \mathbb{T}$ we have $\mathbf{enum}(\#\sigma) =< 6, \#(0 \to \sigma) >$. Let $\sigma \in \mathbb{T}$. Then $\mathbf{enum}(\#\sigma) =< 6, \#(0 \to \sigma) > \in \mathbf{Ix}(0 \to \sigma)$. Let $e \in \mathbf{Ix}(\sigma)$ and $\vec{a} \in (\mathfrak{M})_\sigma$. Let $m \in \omega$.

(i) Suppose $\{\mathbf{enum}(\#\sigma)\}^{0 \to \sigma}(e, \vec{a}) = m$. Then $(\mathbf{enum}(\#\sigma), (e, \vec{a}), m) \in \Theta$. Because $(\mathbf{enum}(\#\sigma))_0 = 6$ we have that $(\mathbf{enum}(\#\sigma), (e, \vec{a}), m) \in \mathsf{ENUM}(\Theta)$ and so $(e, \vec{a}, m) \in \Theta$. Therefore $\{e\}^\sigma(\vec{a}) = m$.

(ii) Now suppose $\{e\}^\sigma(\vec{a}) = m$. Then $(e, \vec{a}, m) \in \Theta$, so $(\mathbf{enum}(\#\sigma), (e, \vec{a}), m) \in \mathsf{ENUM}(\Theta) \subseteq \Phi(\Theta) = \Theta$ and therefore $\{\mathbf{enum}(\#\sigma)\}^{0 \to \sigma}(e, \vec{a}) = m$.

From (i) and (ii) it follows that

$$\{\mathbf{enum}(\#\sigma)\}^{0 \to \sigma}(e, \vec{a}) = \{e\}^\sigma(\vec{a}). \ \square$$

4.5.34. THEOREM. *$\mathcal{R}$ is a recursion theory on* **MON**.

PROOF. By 4.5.27-4.5.33. $\square$

4.5.35. COROLLARY. (i) **MON** *is a special ω-structure.*

(ii) **MON** *is a pre-ω-$\lambda\mathsf{E}\eta(\mathcal{N})$-model.* $\square$

In section 4.3 we proved the First Recursion Theorem for recursion theories on ω-structures in general. In the following we will prove a strong version of the First Recursion Theorem for the recursion theory $\mathcal{R}$ on **MON**: we are going to show that the least fixed point operators $\mathbb{Y}_\sigma$ of **MON** are all $\mathcal{R}$-recursive.

4.5.36. DEFINITION. (i) Define the norm $| \ | : \Theta \to \mathbf{On}$ by

$$|(e, \vec{a}, m)| = \mu\zeta[\, (e, \vec{a}, m) \in \Phi^\zeta \,].$$

(ii) For all $(e, \vec{a}, m) \in \Theta$ we write $|e, \vec{a}, m| = |(e, \vec{a}, m)|$. $\square$

The properties of this norm that we will need are listed in the following lemma.

4.5.37. LEMMA. (i) *Let $\sigma, \tau \in \mathbb{T}$ with $\tau = \tau_1 \to \ldots \to \tau_l \to 0$, $e_0, \ldots, e_l \in \omega$, $\vec{a} \in (\mathfrak{M})_\sigma$, $m \in \omega$. Define $e = \mathbf{sub}(\#\sigma, \#\tau, < e_0, \ldots, e_l >)$. If $(e, \vec{a}, m) \in \Theta$, then there exists $\vec{b} \in (\mathfrak{M})_\tau$ such that*

(1) $(e_0, (\vec{a}, \vec{b}), m) \in \Theta$ and $|e_0, (\vec{a}, \vec{b}), m| < |e, \vec{a}, m|$

(2) for every $j \in \{1, \ldots, l\}$ and all $\vec{c} \in (\mathfrak{M})_{\tau_j}$ with $b_j\vec{c} \in \omega$ we have

$$(e_j, (\vec{a}, \vec{c}), b_j\vec{c}) \in \Theta$$

$$|e_j, (\vec{a}, \vec{c}), b_j\vec{c}| < |e, \vec{a}, m|.$$

(ii) *Let* $\sigma \in \mathbb{T}$ *with* $\sigma = \sigma_1 \to \ldots \to \sigma_k \to 0$, $\vec{s} = (s_1, \ldots, s_l)$ *with* $1 \le s_1, \ldots, s_l \le k$, $e \in \omega$, $\vec{a} \in (\mathfrak{M})_\sigma$, $m \in \omega$. *Define* $e_0 = \mathbf{man}(\#\sigma, <\vec{s}>, e)$. *If* $(e_0, \vec{a}, m) \in \Theta$, *then* $(e, (\vec{a})_{\vec{s}}, m) \in \Theta$ *and* $|e, (\vec{a})_{\vec{s}}, m| < |e_0, \vec{a}, m|$.

(iii) *Let* $\sigma \in \mathbb{T}$, $e \in \omega$, $\vec{a} \in (\mathfrak{M})_\sigma$, $m \in \omega$. *Define* $e_0 = \mathbf{enum}(\#\sigma)$. *If* $(e_0, (e, \vec{a}), m) \in \Theta$, *then* $(e, \vec{a}, m) \in \Theta$ *and* $|e, \vec{a}, m| < |e_0, (e, \vec{a}), m|$.

PROOF. First note that if $|e, \vec{a}, m| = \zeta$, then $(e, \vec{a}, m) \in \Phi^\zeta$ and $(e, \vec{a}, m) \notin \Phi^{(\zeta)}$.

(i) Let $(e, \vec{a}, m) \in \Theta$ and $|e, \vec{a}, m| = \zeta$. Then $(e, \vec{a}, m) \in \Phi^\zeta = \Phi(\Phi^{(\zeta)})$. Because $(e)_0 = 4$ we have $(e, \vec{a}, m) \in \mathsf{SUB}(\Phi^{(\zeta)})$. So there exists $\vec{b} \in (\mathfrak{M})_\tau$ with $(e_0, (\vec{a}, \vec{b}), m) \in \Phi^{(\zeta)}$ and for every $j \in \{1, \ldots, l\}$ and all $\vec{c} \in (\mathfrak{M})_{\tau_j}$

$$b_j\vec{c} \in \omega \Rightarrow (e_j, (\vec{a}, \vec{c}), b_j\vec{c}) \in \Phi^{(\zeta)}.$$

So $(e_0, (\vec{a}, \vec{b}), m) \in \Theta$ and $|e_0, (\vec{a}, \vec{b}), m| < \zeta$, and for all $j \in \{1, \ldots, l\}$ and $\vec{c} \in (\mathfrak{M})_{\tau_j}$ with $b_j\vec{c} \in \omega$ we have $(e_j, (\vec{a}, \vec{c}), b_j\vec{c}) \in \Theta$ and $|e_0, (\vec{a}, \vec{c}), b_j\vec{c}| < \zeta$.

(ii) Let $(e_0, \vec{a}, m) \in \Theta$ and $|e_0, \vec{a}, m| = \zeta$. Then $(e_0, \vec{a}, m) \in \Phi^\zeta = \Phi(\Phi^{(\zeta)})$. Because $(e_0)_0 = 5$ we have $(e_0, \vec{a}, m) \in \mathsf{MAN}(\Phi^{(\zeta)})$ and so $(e, (\vec{a})_{\vec{s}}, m) \in \Phi^{(\zeta)}$ and therefore $(e, (\vec{a})_{\vec{s}}, m) \in \Theta$ and $|e, (\vec{a})_{\vec{s}}, m| < \zeta$.

(iii) Let $(e_0, (e, \vec{a}), m) \in \Theta$ and $\zeta = |e_0, (e, \vec{a}), m|$. Then $(e_0, (e, \vec{a}), m) \in \Phi^\zeta = \Phi(\Phi^{(\zeta)})$. Because $(e_0)_0 = 6$ we have $(e_0, (e, \vec{a}), m) \in \mathsf{ENUM}(\Phi^{(\zeta)})$, and so $(e, \vec{a}, m) \in \Phi^{(\zeta)}$. Therefore $(e, \vec{a}, m) \in \Theta$ and $|e, \vec{a}, m| < \zeta$. $\square$

Using the norm $|\ |$ we are going to prove the abovementioned strengthening of the First Recursion Theorem for the recursion theory $\mathcal{R}$ on $\mathbf{MON}$. Recall the following definitions and facts from section 4.3.

Let $\sigma \in \mathbb{T}, \sigma = \sigma_1 \to \ldots \to \sigma_k \to 0$, and define $\tau = 0 \to (\sigma \to \sigma) \to \sigma$. Now define functionals ψ_σ, χ_σ and φ_σ as follows. Let $\psi_\sigma : \mathfrak{M}_0 \times \mathfrak{M}_{\sigma\to\sigma} \times (\mathfrak{M})_\sigma \times \mathfrak{M}_\sigma \to \mathfrak{M}_0$ be defined by $\psi_\sigma(e, f, \vec{a}, b) = fb\vec{a}$. Let $\chi_\sigma : \mathfrak{M}_0 \times \mathfrak{M}_{\sigma\to\sigma} \times (\mathfrak{M})_\sigma \times (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ be defined by $\chi_\sigma(e, f, \vec{a}, \vec{c}) = \mathcal{E}_\tau(e, e, f, \vec{c})$. Then define $\varphi_\sigma = \mathsf{Sub}_{\tau,\sigma\to 0}[\psi_\sigma, \chi_\sigma]$. Then $\varphi_\sigma : \mathfrak{M}_0 \times \mathfrak{M}_{\sigma\to\sigma} \times (\mathfrak{M})_\sigma \to \mathfrak{M}_0$. For all $e \in \mathfrak{M}_0$, $f \in \mathfrak{M}_{\sigma\to\sigma}$, $\vec{a} \in (\mathfrak{M})_\sigma$ we have that if $b \in \mathfrak{M}_\sigma$ is such that $b\vec{c} = \mathcal{E}_\tau(e, e, f, \vec{c})$ for all $\vec{c} \in (\mathfrak{M})_\sigma$, then $\varphi_\sigma(e, f, \vec{a}) = fb\vec{a}$. We also have that φ_σ is $\mathcal{R}$-recursive, so there exists $e_0 \in \mathbf{Ix}(\tau)$ with $\varphi_\sigma = \{e_0\}^\tau$. If we now define $\Phi_\sigma : \mathfrak{M}_{\sigma\to\sigma} \times (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ by $\Phi_\sigma(f, \vec{a}) = \varphi_\sigma(e_0, f, \vec{a})$, then Φ_σ is $\mathcal{R}$-recursive, and there exists an $\mathcal{R}$-recursive object $\mathbf{y}_\sigma \in \mathfrak{M}_{(\sigma\to\sigma)\to\sigma}$ such that $fnl(\mathbf{y}_\sigma) = \Phi_\sigma$. Then $\mathbf{y}_\sigma f = f(\mathbf{y}_\sigma f)$ for all $f \in \mathfrak{M}_{\sigma\to\sigma}$.

In the following it will be shown that e_0 can be chosen in such a way that $\mathbf{y}_\sigma f$ is the least fixed point of f, for every $f \in \mathfrak{M}_{\sigma\to\sigma}$.

4.5.38. DEFINITION. (i) Let $e_1 \in \mathbf{Ix}(\tau * (\sigma \to 0))$ be such that $\psi_\sigma = \{e_1\}^{\tau*(\sigma\to 0)}$.

(ii) $e_2 = \mathbf{man}(\#(\tau * \sigma), < \vec{s} >, \mathbf{enum}(\#\tau))$ where $\vec{s}$ is defined by

$$\vec{s} = (1, 1, 2, k+3, k+4, k+5, \ldots, 2k+2).$$

(iii) $e_0 = \mathbf{sub}(\#\tau, \#(\sigma \to 0), < e_1, e_2 >)$. $\square$

4.5.39. LEMMA. (i) *$e_2 \in \mathbf{Ix}(\tau * \sigma)$ and $\chi_\sigma = \{e_2\}^{\tau * \sigma}$.*
(ii) *$e_0 \in \mathbf{Ix}(\tau)$ and $\varphi_\sigma = \{e_0\}^{\tau}$.*

PROOF. Simple. $\square$

4.5.40. PROPOSITION. *Let $e \in \mathbf{Ix}(\tau)$, $f \in \mathfrak{M}_{\sigma \to \sigma}$, $\vec{a} \in (\mathfrak{M})_\sigma$ and $m \in \omega$ be such that $(e_0, (e, f, \vec{a}), m) \in \Theta$. Then there exists $b \in \mathfrak{M}_\sigma$ such that $m = fb\vec{a}$ and for all $\vec{c} \in (\mathfrak{M})_\sigma$ with $b\vec{c} \in \omega$ we have $(e, (e, f, \vec{c}), b\vec{c}) \in \Theta$ and $|e, (e, f, \vec{c}), b\vec{c}| < |e_0, (e, f, \vec{a}), m|$.*

PROOF. Define $\zeta = |e_0, (e, f, \vec{a}), m|$. Recall that

$$e_0 = \mathbf{sub}(\#\tau, \#(\sigma \to 0), < e_1, e_2 >)$$

so by lemma 4.5.37(i) there exists $b \in \mathfrak{M}_\sigma$ such that
(i) $(e_1, (e, f, \vec{a}, b), m) \in \Theta$
(ii) for all $\vec{c} \in (\mathfrak{M})_\sigma$ with $b\vec{c} \in \omega$ we have

$$(e_2, (e, f, \vec{a}, \vec{c}), b\vec{c}) \in \Theta$$

$$|e_2, (e, f, \vec{a}, \vec{c}), b\vec{c}| < \zeta.$$

Then by (i) we have

$$\begin{aligned} m &= \{e_1\}^{\tau * (\sigma \to 0)}(e, f, \vec{a}, b) \\ &= \psi_\sigma(e, f, \vec{a}, b) \\ &= fb\vec{a}. \end{aligned}$$

Now let $\vec{c} \in (\mathfrak{M})_\sigma$ be such that $b\vec{c} \in \omega$. Then by (ii) we have

$$(e_2, (e, f, \vec{a}, \vec{c}), b\vec{c}) \in \Theta$$

$$|e_2, (e, f, \vec{a}, \vec{c}), b\vec{c}| < \zeta.$$

Because $e_2 = \mathbf{man}(\#(\tau * \sigma), < \vec{s} >, \mathbf{enum}(\#\tau))$ this gives by lemma 4.5.37(ii)

$$(\mathbf{enum}(\#\tau), (e, e, f, \vec{c}), b\vec{c}) \in \Theta$$

$$|\mathbf{enum}(\#\tau), (e, e, f, \vec{c}), b\vec{c}| < |e_2, (e, f, \vec{a}, \vec{c}), b\vec{c}|.$$

Then by lemma 4.5.37(iii) it follows that

$$(e, (e, f, \vec{c}), b\vec{c}) \in \Theta$$

$$|e,(e,f,\vec{c}),b\vec{c}| < |\mathbf{enum}(\#\tau),(e,e,f,\vec{c}),b\vec{c}|.$$

Therefore we have

$$\begin{aligned} |e,(e,f,\vec{c}),b\vec{c}| &< |\mathbf{enum}(\#\tau),(e,e,f,\vec{c}),b\vec{c}| \\ &< |e_2,(e,f,\vec{a},\vec{c}),b\vec{c}| \\ &< \zeta \\ &= |e_0,(e,f,\vec{a}),m|. \ \Box \end{aligned}$$

4.5.41. Definition. (i) For every $\zeta \in \mathbf{On}$ define

$$\Theta_\zeta = \{(e,\vec{a},m) \in \Theta \mid |e,\vec{a},m| < \zeta\}.$$

(ii) For all $e \in \omega$, $\sigma \in \mathbb{T}$ and $\zeta \in \mathbf{On}$ define $\{e\}^\sigma_\zeta : (\mathfrak{M})_\sigma \to \omega_\perp$ by

(1) $\{e\}^\sigma_\zeta(\vec{a}) = \{e\}^\sigma(\vec{a})$ if $(e,\vec{a},\{e\}^\sigma(\vec{a})) \in \Theta_\zeta$

(2) $\{e\}^\sigma_\zeta(\vec{a}) = \perp$ otherwise. $\Box$

For all $e \in \omega$, $\sigma \in \mathbb{T}$, $\zeta \in \mathbf{On}$, $\vec{a} \in (\mathfrak{M})_\sigma$ and $m \in \omega$

$$\{e\}^\sigma_\zeta(\vec{a}) = m \iff (e,\vec{a},m) \in \Theta_\zeta.$$

4.5.42. Proposition. *Let $f \in \mathfrak{M}_{\sigma\to\sigma}$. Let $d \in \mathfrak{M}_\sigma$ be such that $fd = d$.*

(i) *For all $\zeta \in \mathbf{On}$, $\vec{a} \in (\mathfrak{M})_\sigma$, $m \in \omega$*

$$\{e_0\}^\tau_\zeta(e_0,f,\vec{a}) = m \Rightarrow m = d\vec{a}.$$

(ii) *$\{e_0\}^\tau(e_0,f,\vec{a}) \sqsubseteq d\vec{a}$ for all $\vec{a} \in (\mathfrak{M})_\sigma$.*

Proof. (i) By transfinite induction on ζ. Let $\zeta \in \mathbf{On}$ and suppose that for all $\xi < \zeta$ and for all $\vec{a} \in (\mathfrak{M})_\sigma$ and $m \in \omega$

$$\{e_0\}^\tau_\xi(e_0,f,\vec{a}) = m \Rightarrow m = d\vec{a}.$$

Now let $\vec{a} \in (\mathfrak{M})_\sigma$ and $m \in \omega$ be such that

$$\{e_0\}^\tau_\zeta(e_0,f,\vec{a}) = m.$$

Then $(e_0,(e_0,f,\vec{a}),m) \in \Theta_\zeta$, so $(e_0,(e_0,f,\vec{a}),m) \in \Theta$ and $|e_0,(e_0,f,\vec{a}),m| < \zeta$. By the previous proposition there exists $b \in \mathfrak{M}_\sigma$ such that $m = fb\vec{a}$ and for all $\vec{c} \in (\mathfrak{M})_\sigma$ with $b\vec{c} \in \omega$

$$(e_0,(e_0,f,\vec{c}),b\vec{c}) \in \Theta$$

$$|e_0,(e_0,f,\vec{c}),b\vec{c}| < |e_0,(e_0,f,\vec{a}),m|.$$

Define $\xi = |e_0, (e_0, f, \vec{a}), m|$. Then $\xi < \zeta$. Let $\vec{c} \in (\mathfrak{M})_\sigma$ and suppose $b\vec{c} \in \omega$. Then $(e_0, (e_0, f, \vec{c}), b\vec{c}) \in \Theta_\zeta$, so

$$\{e_0\}^\tau_\xi(e_0, f, \vec{c}) = b\vec{c} \in \omega.$$

By induction this gives $b\vec{c} = d\vec{c}$. So for all $\vec{c} \in (\mathfrak{M})_\sigma$ we have $b\vec{c} \in \omega \Rightarrow b\vec{c} = d\vec{c}$, and so $b \sqsubseteq d$. Because f is monotone we have $m = fb\vec{a} \sqsubseteq fd\vec{a} = d\vec{a}$, because $fd = d$. But $m \in \omega$ and so $m = d\vec{a}$. This proves (i).

(ii) Let $\vec{a} \in (\mathfrak{M})_\sigma$ and suppose

$$\{e_0\}^\tau(e_0, f, \vec{a}) = m \in \omega.$$

Then $(e_0, (e_0, f, \vec{a}), m) \in \Theta$. Define

$$\zeta = |e_0, (e_0, f, \vec{a}), m| + 1.$$

Then $\{e_0\}^\tau_\zeta(e_0, f, \vec{a}) = m$, so by (i) we have $m = d\vec{a}$. Therefore $\{e_0\}^\tau(e_0, f, \vec{a}) \sqsubseteq d\vec{a}$. □

Define $\Phi_\sigma : \mathfrak{M}_{\sigma\to\sigma} \times (\mathfrak{M})_\sigma \to \mathfrak{M}_0$ by $\Phi_\sigma(f, \vec{a}) = \varphi_\sigma(e_0, f, \vec{a})$. Let $\mathbf{y}_\sigma \in \mathfrak{M}_{(\sigma\to\sigma)\to\sigma}$ be such that $fnl(\mathbf{y}_\sigma) = \Phi_\sigma$. By the proof of the First Recursion Theorem in section 4.3 we have that $\mathbf{y}_\sigma$ is an $\mathcal{R}$-recursive object with the property that $\mathbf{y}_\sigma f = f(\mathbf{y}_\sigma f)$ for all $f \in \mathfrak{M}_{\sigma\to\sigma}$. Below it will be shown that $\mathbf{y}_\sigma$ gives least fixed points, so that $\mathbf{y}_\sigma = \mathbb{Y}_\sigma$.

4.5.43. PROPOSITION. (i) *Let* $f \in \mathfrak{M}_{\sigma\to\sigma}$. *Let* $d \in \mathfrak{M}_\sigma$ *be such that* $fd = d$. *Then* $\mathbf{y}_\sigma f \sqsubseteq d$.

(ii) $\mathbf{y}_\sigma = \mathbb{Y}_\sigma$.

PROOF. (i) For all $\vec{a} \in (\mathfrak{M})_\sigma$

$$\begin{aligned} \mathbf{y}_\sigma f\vec{a} &= \Phi_\sigma(f, \vec{a}) \\ &= \varphi_\sigma(e_0, f, \vec{a}) \\ &= \{e_0\}^\tau(e_0, f, \vec{a}) \\ &\sqsubseteq d\vec{a}. \end{aligned}$$

Therefore $\mathbf{y}_\sigma f \sqsubseteq d$.

(ii) By (i) we have that $\mathbf{y}_\sigma f$ is the least fixed point of f, for all $f \in \mathfrak{M}_{\sigma\to\sigma}$, so $\mathbf{y}_\sigma f = \mathbb{Y}_\sigma f$. Therefore $\mathbf{y}_\sigma = \mathbb{Y}_\sigma$, because **MON** is extensional. □

4.5.44. THEOREM. *The least fixed point operators* $\mathbb{Y}_\sigma$ *of* **MON** *are all* $\mathcal{R}$*-recursive.* □

4.5.45. COROLLARY. *Every* $\lambda\mathsf{Y}(\mathcal{N})$*-definable object of* **MON** *is* $\mathcal{R}$*-recursive.* □

It is an open problem whether or not there exist $\mathcal{R}$-recursive objects in **MON** that are not $\lambda\mathsf{Y}(\mathcal{N})$-definable.

Chapter 5

Diagrams of type structures

The diagram of a type structure $\mathfrak{M}$ is a system $\lambda(\mathfrak{M})$ which contains a representation of the structure $\mathfrak{M}$ in the form of constants $\mathbf{C}(a)$ for all objects a of $\mathfrak{M}$. The connection between this chapter and the previous chapters (especially chapter 2) is the notion of quasi-extensionality. The main result of this chapter is the characterization of the class of all quasi-extensional type structures as precisely those type structures $\mathfrak{M}$ for which we have $\lambda(\mathfrak{M}) \models \mathbf{CR}$. From this main result a number of other results follow.

Diagrams are based on a type-free λ-calculus that [Kleene 1962] used in order to introduce a notion of computability for partial functionals of higher types over the natural numbers.

5.1. Definitions and basic properties.

For every (separated) type structure $\mathfrak{M}$ we are going to construct its diagram $\lambda(\mathfrak{M})$. The diagram $\lambda(\mathfrak{M})$ will be a $\beta\delta$-system; the δ-reduction of $\lambda(\mathfrak{M})$ will be called μ-reduction.

Let $\mathfrak{M}$ be a type structure.

5.1.1. DEFINITION. For each $a \in |\mathfrak{M}|$ take a constant $\mathbf{C}(a)$.

(i) For each $\sigma \in \mathbb{T}$ the set $\mathcal{T}_\sigma(\mathfrak{M})$ of $\lambda(\mathfrak{M})$-terms of type σ is defined inductively as follows.

(1) $Var_\sigma \subseteq \mathcal{T}_\sigma(\mathfrak{M})$.

(2) $a \in \mathfrak{M}_\sigma \Rightarrow \mathbf{C}(a) \in \mathcal{T}_\sigma(\mathfrak{M})$.

(3) $M \in \mathcal{T}_{\sigma\to\tau}(\mathfrak{M})\ \&\ N \in \mathcal{T}_\sigma(\mathfrak{M}) \Rightarrow (MN) \in \mathcal{T}_\tau(\mathfrak{M})$.

(4) $x \in Var_\sigma\ \&\ M \in \mathcal{T}_\tau(\mathfrak{M}) \Rightarrow (\lambda x.M) \in \mathcal{T}_{\sigma\to\tau}(\mathfrak{M})$.

(ii) The set of all $\lambda(\mathfrak{M})$-terms is

$$\mathcal{T}(\mathfrak{M}) = \bigcup_{\sigma\in\mathbb{T}} \mathcal{T}_\sigma(\mathfrak{M}). \ \square$$

Note that $\mathcal{T}(\mathfrak{M}) = \mathcal{T}(\mathbb{C})$ where $\mathbb{C} = (\mathbb{C}_\sigma)_{\sigma\in\mathbb{T}}$ with $\mathbb{C}_\sigma = \{\mathbf{C}(a) | a \in \mathfrak{M}_\sigma\}$ for every $\sigma \in \mathbb{T}$.

In the following we are going to define μ-reduction on $\mathcal{T}(\mathfrak{M})$. The diagram of $\mathfrak{M}$ will then be $\lambda(\mathfrak{M}) = (\mathcal{T}(\mathfrak{M}), \beta\mu)$.

The idea is that (closed) $\lambda(\mathfrak{M})$-terms are programs with oracles. These programs compute objects of $\mathfrak{M}$ from other objects. Suppose that we have a reduction $\to$ (extending $\to_\beta$) on $\mathcal{T}(\mathfrak{M})$, that gives a conversion $=$ on $\mathcal{T}(\mathfrak{M})$. Since each object a has a representative $\mathbf{C}(a)$, we can use the conversion $=$ to compute objects by $\lambda(\mathfrak{M})$-terms as follows. If $\sigma = \sigma_1 \to \cdots \to \sigma_k \to 0 \in \mathbb{T}$, $M \in \mathcal{T}_\sigma(\mathfrak{M})^0$ and $a \in \mathfrak{M}_\sigma$, then M computes a if for all $\vec{b} \in (\mathfrak{M})_\sigma$

$$M\mathbf{C}(b_1)\ldots\mathbf{C}(b_k) = \mathbf{C}(ab_1\ldots b_k).$$

Kleene's idea behind the reduction in his 1962 paper is that type-0 objects of $\mathfrak{M}$ are values (outputs), and that higher type objects, or rather: their constants, are oracles. The oracle $\mathbf{C}(a)$ (with $a \in \mathfrak{M}_\sigma$ and $\sigma = \sigma_1 \to \cdots \to \sigma_k \to 0 \neq 0$) takes as arguments closed $\lambda(\mathfrak{M})$-terms (programs) $M_1, \ldots, M_k$. ([Kleene 1962] puts an extra restriction on these terms, but we won't.) If these programs compute objects $b_1, \ldots, b_k$, then $\mathbf{C}(a)$ acting on $M_1, \ldots, M_k$ returns the value $ab_1 \ldots b_k$ in the form of the constant $\mathbf{C}(ab_1 \ldots b_k)$. This is formalized by adding as an extra reduction rule

$$\mathbf{C}(a)M_1\ldots M_k \to \mathbf{C}(ab_1\ldots b_k)$$

which expresses the behaviour of $\mathbf{C}(a)$ as an oracle. Of course, this extends the reduction and conversion on $\mathcal{T}(\mathfrak{M})$, and therefore this also extends the notion of computability, which in turn forces us, by the oracle intuition above, to add even more new reduction rules. This process eventually reaches a fixed point, which is the reductor μ.

In the following we will give a formal construction of μ, using an inductive definition. Let $\mathbb{R}$ be the set of all reductors on $\mathcal{T}(\mathfrak{M})$. If for every $i \in I$ we have a reductor $\mathbf{R}_i$ on $\mathcal{T}(\mathfrak{M})$, then their union $\bigcup\{\mathbf{R}_i | i \in I\}$ is also a reductor on $\mathcal{T}(\mathfrak{M})$. From this it follows that $(\mathbb{R}, \subseteq)$ is a cpo. So every monotone operator $\Phi : \mathbb{R} \to \mathbb{R}$ has a least fixed point $\Phi^\infty \in \mathbb{R}$.

5.1.2. Definition. Let $\mathbf{R} \in \mathbb{R}$. The relation of R-computability, written as $\rhd_R$, between $\mathcal{T}(\mathfrak{M})^0$ and $|\mathfrak{M}|$ is defined as follows. Let $\sigma \in \mathbb{T}$, $M \in \mathcal{T}_\sigma(\mathfrak{M})^0$, $a \in \mathfrak{M}_\sigma$. Then M R-computes a (notation $M \rhd_R a$) if for all $(b_1, \ldots, b_k) \in (\mathfrak{M})_\sigma$

$$M\mathbf{C}(b_1)\ldots\mathbf{C}(b_k) =_{\beta R} \mathbf{C}(ab_1\ldots b_k). \ \square$$

By convention we have that for all $M \in \mathcal{T}_0(\mathfrak{M})^0$ and $a \in \mathfrak{M}_\sigma$

$$M \rhd_R a \iff M =_{\beta R} \mathbf{C}(a).$$

Note that R-computability is merely a relation between closed $\lambda(\mathfrak{M})$-terms and objects of $\mathfrak{M}$, and not a notion of computability for objects of $\mathfrak{M}$.

5.1.3. LEMMA. (i) *Let* $\mathbf{R}_1, \mathbf{R}_2 \in \mathbb{R}$ *with* $\mathbf{R}_1 \subseteq \mathbf{R}_2$. *Then* $\rhd_{R_1}$ *is a subrelation of* $\rhd_{R_2}$.

(ii) *Let* $\mathbf{R} \in \mathbb{R}$. *Let* $\sigma \in \mathbb{T}$, $M \in \mathcal{T}_\sigma(\mathfrak{M})^0$, $a, b \in \mathfrak{M}_\sigma$. *Then*

$$M \rhd_R a \;\&\; a \approx b \Rightarrow M \rhd_R b.$$

PROOF. Simple. □

5.1.4. DEFINITION. Let $\mathbf{R} \in \mathbb{R}$. Then $\Phi(\mathbf{R})$ is the reductor consisting of all reduction rules

$$\mathbf{C}(a) M_1 \ldots M_k \to \mathbf{C}(ab_1 \ldots b_k)$$

with for $i = 1, \ldots, k$

$$M_i \rhd_R b_i$$

where $\sigma = \sigma_1 \to \cdots \to \sigma_k \to 0 \neq 0$, $a \in \mathfrak{M}_\sigma$, $(b_1, \ldots, b_k) \in (\mathfrak{M})_\sigma$ and $M_i \in \mathcal{T}_{\sigma_i}(\mathfrak{M})^0$ $(i = 1, \ldots, k)$. □

By this definition we have an operator $\Phi : \mathbb{R} \to \mathbb{R}$. If $\mathbf{R}_1, \mathbf{R}_2 \in \mathbb{R}$ with $\mathbf{R}_1 \subseteq \mathbf{R}_2$, then $\rhd_{R_1}$ is a subrelation of $\rhd_{R_2}$ and therefore $\Phi(\mathbf{R}_1) \subseteq \Phi(\mathbf{R}_2)$. So Φ is monotone, and therefore has a least fixed point $\Phi^\infty \in \mathbb{R}$.

5.1.5. DEFINITION. (i) $\mu = \Phi^\infty$.

(ii) The diagram of $\mathfrak{M}$ is the system

$$\lambda(\mathfrak{M}) = (\mathcal{T}(\mathfrak{M}), \beta\mu).$$

(iii) The relation $\rhd_\mu$ of μ-computability is called $\lambda(\mathfrak{M})$-computability and is denoted by $\rhd$. □

So, for all $\sigma \in \mathbb{T}$, $M \in \mathcal{T}_\sigma(\mathfrak{M})^0$ and $a \in \mathfrak{M}_\sigma$ we have that M $\lambda(\mathfrak{M})$-computes a (written as $M \rhd a$) if for all $\vec{b} \in (\mathfrak{M})_\sigma$

$$M\mathbf{C}(\vec{b}) =_{\beta\mu} \mathbf{C}(a\vec{b}).$$

Because $\mu = \Phi(\mu)$, the reductor μ consists of all reduction rules

$$\mathbf{C}(a) M_1 \ldots M_k \to \mathbf{C}(ab_1 \ldots b_k)$$

with for $i = 1, \ldots, k$

$$M_i \rhd b_i$$

where $\sigma = \sigma_1 \to \cdots \to \sigma_k \to 0 \neq 0$, $a \in \mathfrak{M}_\sigma$, $(b_1, \ldots, b_k) \in (\mathfrak{M})_\sigma$ and $M_i \in \mathcal{T}_{\sigma_i}(\mathfrak{M})^0$ $(i = 1, \ldots, k)$. It follows that μ is a δ-reductor and that $\lambda(\mathfrak{M})$ is a $\beta\delta$-system.

5.1.6. LEMMA. *Let $\sigma \in \mathbb{T}$.*

(i) *If $\sigma \neq 0$, then for all $a \in \mathfrak{M}_\sigma$ and $\vec{b} \in (\mathfrak{M})_\sigma$*

$$\mu : \mathbf{C}(a)\mathbf{C}(\vec{b}) \to \mathbf{C}(ab_1 \dots b_k).$$

(ii) *For all $a \in \mathfrak{M}_\sigma$*

$$\mathbf{C}(a) \triangleright a.$$

PROOF. By induction on the structure of σ. For $\sigma = 0$ the result is trivial. Let $\sigma = \sigma_1 \to \dots \to \sigma_k \to 0$ with $\sigma \neq 0$.

(i) Let $a \in \mathfrak{M}_\sigma$, $\vec{b} = (b_1, \dots, b_k) \in (\mathfrak{M})_\sigma$. By induction $\mathbf{C}(b_i) \triangleright b_i$ ($i = 1, \dots, k$), and therefore

$$\mu : \mathbf{C}(a)\mathbf{C}(b_1) \dots \mathbf{C}(b_k) \to \mathbf{C}(a\vec{b}).$$

(ii) Let $a \in \mathfrak{M}_\sigma$. By (i) we have that for all $(b_1, \dots, b_k) \in (\mathfrak{M})_\sigma$

$$\mathbf{C}(a)\mathbf{C}(b_1) \dots \mathbf{C}(b_k) \to \mathbf{C}(ab_1 \dots b_k)$$

so $\mathbf{C}(a)\mathbf{C}(\vec{b}) =_{\beta\mu} \mathbf{C}(a\vec{b})$, and therefore $\mathbf{C}(a) \triangleright a$. $\Box$

A natural quesion to ask is for which type structures $\mathfrak{M}$ we have $\lambda(\mathfrak{M}) \models \mathbf{CR}$. Below it will be shown that if $\lambda(\mathfrak{M}) \models \mathbf{CR}$, then $\mathfrak{M}$ is quasi-extensional. In section 5.3 it will be shown that quasi-extensionality of $\mathfrak{M}$ is also sufficient for $\lambda(\mathfrak{M}) \models \mathbf{CR}$.

5.1.7. LEMMA. *If $\lambda(\mathfrak{M}) \models \mathbf{CR}$, then $\mathfrak{M}$ is quasi-extensional.*

PROOF. Suppose $\lambda(\mathfrak{M}) \models \mathbf{CR}$. Let $\sigma \in \mathbb{T}$ and $a \in \mathfrak{M}_\sigma$. Let $\vec{b}, \vec{c} \in (\mathfrak{M})_\sigma$ with $\vec{b} \approx \vec{c}$. Since $\mathbf{C}(b_i) \triangleright b_i$ for each i, also $\mathbf{C}(b_i) \triangleright c_i$ because $b_i \approx c_i$. This gives $\mathbf{C}(a)\mathbf{C}(\vec{b}) \twoheadrightarrow_\mu \mathbf{C}(a\vec{b})$ and $\mathbf{C}(a)\mathbf{C}(\vec{b}) \twoheadrightarrow_\mu \mathbf{C}(a\vec{c})$. By $\mathbf{CR}(\beta\mu)$ and the fact that $\mathbf{C}(a\vec{b})$ and $\mathbf{C}(a\vec{c})$ are $\beta\mu$-normal forms, we have $\mathbf{C}(a\vec{b}) \equiv \mathbf{C}(a\vec{c})$ and so $a\vec{b} = a\vec{c}$. Therefore a is extensional. So $\mathfrak{M}$ is quasi-extensional. $\Box$

Because μ is not $\beta\mu$-normal, we cannot apply the First Church-Rosser Theorem in proving $\lambda(\mathfrak{M}) \models \mathbf{CR}$ for quasi-extensional $\mathfrak{M}$. In the next section we will prove a stronger Church-Rosser theorem for $\beta\delta$-reduction, that we will use in section 5.3 in order to prove $\lambda(\mathfrak{M}) \models \mathbf{CR}$ for quasi-extensional $\mathfrak{M}$.

5.2. More on delta-reduction.

In this section we look at a few technical tools that we need in the following sections. We prove a Church-Rosser theorem for $\beta\delta$-reduction, and we define residuals of constants and of δ-redexes.

We start with the Church-Rosser theorem. For this we need the notion of resistant δ-reductor.

5.2.1. DEFINITION. Let δ be a δ-reductor on $\mathcal{T}(\mathbb{C})$, and let $\mathbf{R}$ be any reductor on $\mathcal{T}(\mathbb{C})$. Then δ is called R-resistant if for every δ-rule

$$\delta : \mathbf{C}M_1 \dots M_k \to N$$

and for all $N_1, \dots, N_k \in \mathcal{T}(\mathbb{C})$ with $M_i \twoheadrightarrow_R N_i$ $(i = 1, \dots, k)$ we have that

$$\delta : \mathbf{C}N_1 \dots N_k \to N. \ \square$$

So δ is $\mathbf{R}$-resistant if every δ-redex remains a δ-redex after internal R-reduction, and moreover has the same contractum.

Note that δ is $\mathbf{R}$-resistant if and only if δ can resist internal one-step R-reductions: suppose that for every δ-rule $\mathbf{C}M_1 \dots M_k \to N$, every $i \in \{1, \dots, k\}$, all $N_1, \dots, N_k \in \mathcal{T}(\mathbb{C})$ with

$$M_i \to_R N_i \ \& \ \forall j \neq i [\, M_j \equiv N_j \,]$$

we have $\delta : \mathbf{C}N_1 \dots N_k \to N$; then δ is R-resistant.

If $\mathbf{R}_1$ and $\mathbf{R}_2$ are reductors on $\mathcal{T}(\mathbb{C})$, and δ is $\mathbf{R}_1$-resistant and $\mathbf{R}_2$-resistant, then δ is also $\mathbf{R}_1\mathbf{R}_2$-resistant.

The next theorem is a strengthening of the First Church-Rosser Theorem for $\beta\delta$-reduction, and is a simply typed variant of Mitschke's Theorem on δ-reduction in the type-free λ-calculus.

5.2.2. THEOREM (Second Church-Rosser Theorem).
Let δ be a δ-reductor on $\mathcal{T}(\mathbb{C})$ such that
(i) *δ is closed*
(ii) *δ is deterministic*
(iii) *δ is $\beta\delta$-resistant.*
Then $\mathbf{CR}(\beta\delta)$.

PROOF. We proceed in three steps.

(i) First we prove that β and δ commute. Let $M \to_\beta^{O_1} M_1$ and $M \to_\delta^{O_2} M_2$ with $O_1 = (C_1, (\lambda x.A)B)$ and $O_2 = (C_2, \mathbf{C}P_1 \dots P_m)$. Say $\delta : \mathbf{C}P_1 \dots P_m \to P$ and $M_2 \equiv C_2[P]$. Distinguish the following three cases.

(1) O_1 and O_2 are disjoint. Then there trivially exists $N \in \mathcal{T}(\mathbb{C})$ with $M_1 \to_\delta N$ and $M_2 \to_\beta N$.

(2) $O_1 \subseteq O_2$. Then O_1 is in P_i for some $i \in \{1, \ldots, m\}$. Say $P_i \to_\beta Q_i$ by contracting $(\lambda x.A)B$. For $j \neq i$ define $Q_j \equiv P_j$. Then $M_1 \equiv C_2[\mathbf{C}Q_1 \ldots Q_m]$. Because δ is β-resistant it follows that $\delta : \mathbf{C}Q_1 \ldots Q_m \to P$ and so $M_1 \to_\delta M_2$.

(3) $O_2 \subseteq O_1$. There are two subcases:

- O_2 is in A. Say $A \to_\delta A'$ by contracting $\mathbf{C}P_1 \ldots P_m$. Then we have $A[x := B] \to_\delta A'[x := B]$, so

$$\begin{tikzcd}
M \equiv C_1[(\lambda x.A)B] \arrow[r, "\delta"] \arrow[d, "\beta"'] & C_1[(\lambda x.A')B] \equiv M_2 \arrow[d, "\beta"] \\
M_1 \equiv C_1[A[x := B]] \arrow[r, "\delta"'] & C_1[A'[x := B]]
\end{tikzcd}$$

- O_2 is in B. Say $B \to_\delta B'$ by contracting $\mathbf{C}P_1 \ldots P_m$. Then we have $A[x := B] \twoheadrightarrow_\delta A[x := B']$, so

$$\begin{tikzcd}
M \equiv C_1[(\lambda x.A)B] \arrow[r, "\delta"] \arrow[d, "\beta"'] & C_1[(\lambda x.A)B'] \equiv M_2 \arrow[d, "\beta"] \\
M_1 \equiv C_1[A[x := B]] \arrow[r, two heads, "\delta"] & C_1[A[x := B']]
\end{tikzcd}$$

In any of the cases we have

$$\begin{tikzcd}
M \arrow[r, "\delta"] \arrow[d, "\beta"'] & M_2 \arrow[d, "\equiv", "\beta"'] \\
M_1 \arrow[r, two heads, "\delta"] & N
\end{tikzcd}$$

for some $N \in \mathcal{T}(\mathbb{C})$. Therefore β and δ commute.

(ii) Now we prove $\mathbf{CR}(\delta)$. Let $M \to_\delta^{O_1} M_1$ and $M \to_\delta^{O_2} M_2$ with $O_1 = (C_1, \mathbf{C}_1 P_1 \ldots P_m)$ and $O_2 = (C_2, \mathbf{C}_2 Q_1 \ldots Q_n)$. Say $\delta : \mathbf{C}_1 P_1 \ldots P_m \to P$, $\delta : \mathbf{C}_2 Q_1 \ldots Q_n \to Q$, $M_1 \equiv C_1[P]$ and $M_2 \equiv C_2[Q]$. Now distinguish four cases.

(1) O_1 and O_2 are disjoint. Then there trivially exists $N \in \mathcal{T}(\mathbb{C})$ with $M_1 \to_\delta N$ and $M_2 \to_\delta N$.

(2) $O_1 = O_2$. Then $C_1 = C_2$ and $\mathbf{C}_1 P_1 \dots P_m = \mathbf{C}_2 Q_1 \dots Q_n$, so $\mathbf{C}_1 = \mathbf{C}_2$, $m = n$ and $P_i = Q_i$ $(i = 1, \dots, m)$. Because δ is deterministic, it follows that $P \equiv Q$ and so $M_1 \equiv C_1[P] \equiv C_2[Q] \equiv M_2$.

(3) $O_1 \subseteq O_2$ and $O_1 \neq O_2$. Then O_1 is contained in Q_i for some $i \in \{1, \dots, n\}$. Say $Q_i \to_\delta Q'_i$ by contracting O_1. For $j \neq i$ define $Q'_j \equiv Q_j$. Then $M_1 \equiv C_2[\mathbf{C}_2 Q'_1 \dots Q'_n]$. Because δ is δ-resistant we have $\delta : \mathbf{C}_2 Q'_1 \dots Q'_n \to Q$ and so $M_1 \equiv C_2[\mathbf{C}_2 Q'_1 \dots Q'_n] \to_\delta C_2[Q] \equiv M_2$.

(4) $O_2 \subseteq O_1$ and $O_2 \neq O_1$. This case is similar to case (3). Now we have $M_2 \to_\delta M_1$.

In any of these cases we have

$$\begin{tikzcd} M \arrow[r, "\delta"] \arrow[d, "\delta"'] & M_2 \arrow[d, "\equiv"', "\delta"] \\ M_1 \arrow[r, "\equiv", "\delta"'] & N \end{tikzcd}$$

for some $N \in \mathcal{T}(\mathbb{C})$. Therefore $\mathbf{CR}(\delta)$.

(iii) Using the Hindley-Rosen Lemma it follows from (i), (ii) and $\mathbf{CR}(\beta)$ that $\beta\delta$ has the Church-Rosser Property. □

In the following we are going to define residuals of constants and of δ-redexes relative to $\beta\delta$-reduction sequences. For this we adapt the definition of residuals of β-redexes in section 11.2 of [Barendregt 1984].

First we review subterm occurrences. A subterm occurrence in M is a pair $O = (C, N)$ with $M = C[N]$. If O is a subterm occurrence in M, then we write $O \in M$. If $\mathcal{F}$ is a set of subterm occurrences in M, then we write $\mathcal{F} \subseteq M$. A constant occurrence is a subterm occurrence of the form $(C, \mathbf{C})$ where $\mathbf{C}$ is a constant.

Next we define reduction sequences. Let $\mathbf{R}$ be a reductor on $\mathcal{T}(\mathbb{C})$. An R-reduction step is a triple (M, N, O) such that $O = (C, P)$, $M \equiv C[P]$ and there exists $Q \in \mathcal{T}(\mathbb{C})$ with $(P, Q) \in \mathbf{R}$ and $N \equiv C[Q]$. An R-reduction step (M, N, O) is usually written as

$$M \to^O N.$$

An R-reduction sequence is a sequence π of consecutive R-reduction steps

$$M_0 \to^{O_0} M_1, M_1 \to^{O_1} M_2, \dots, M_{n-1} \to^{O_{n-1}} M_n.$$

This sequence will be written as

$$\pi : M_0 \to^{O_0} M_1 \to^{O_1} \cdots \to^{O_{n-1}} M_n.$$

The sequence π is said to start with M_0 and to end with M_n; we write $\pi : M_0 \twoheadrightarrow M_n$.

In order to keep track of constant occurrences in a $\beta\delta$-reduction we use marked constants.

5.2.3. DEFINITION. (i) For every $\mathbf{C} \in |\mathbb{C}|$ take a new constant $\tilde{\mathbf{C}}$, called $\mathbf{C}$-mark.
(ii) $\tilde{\mathbb{C}} = (\tilde{\mathbb{C}}_\sigma)_{\sigma \in \mathbb{T}}$ with $\tilde{\mathbb{C}}_\sigma = \mathbb{C}_\sigma \cup \{\tilde{\mathbf{C}} | \mathbf{C} \in \mathbb{C}_\sigma\}$ for every $\sigma \in \mathbb{T}$. $\square$

Next to $\mathcal{T}(\mathbb{C})$ we now also have a set $\mathcal{T}(\tilde{\mathbb{C}})$ of λ-terms in which some constants can be marked. The set $\mathcal{T}(\tilde{\mathbb{C}})$ is an extension of $\mathcal{T}(\mathbb{C})$.

5.2.4. DEFINITION. (i) If $M \in \mathcal{T}(\mathbb{C})$ and $\mathcal{F}$ is a set of constant occurrences in M, then $(M, \mathcal{F}) \in \mathcal{T}(\tilde{\mathbb{C}})$ is obtained from M by marking all constant occurrences that are in $\mathcal{F}$.
(ii) If $M' \in \mathcal{T}(\tilde{\mathbb{C}})$, then $|M'| \in \mathcal{T}(\mathbb{C})$ is obtained from M' by leaving out all markings. $\square$

In the terminology of section 1.5 the mapping $|\ | : \mathcal{T}(\tilde{\mathbb{C}}) \to \mathcal{T}(\mathbb{C})$ is the transformation with the property that $|\tilde{\mathbf{C}}| = \mathbf{C}$ and $|\mathbf{C}| = \mathbf{C}$ for all $\mathbf{C} \in |\mathbb{C}|$.

5.2.5. LEMMA. (i) $|(M, \mathcal{F})| = M$.
(ii) *If $|M'| = M$, then there exists a unique set $\mathcal{F}$ of constant occurrences in M such that $M' = (M, \mathcal{F})$.* $\square$

The truth of this lemma is obvious. A precise proof requires precise definitions of $|M'|$ and $(M, \mathcal{F})$ and will be omitted.

5.2.6. DEFINITION. If $C \in Context(\tilde{\mathbb{C}})$, then $|C| \in Context(\mathbb{C})$ is obtained from C by leaving out all markings. $\square$

By induction on the structure of $C \in Context(\tilde{\mathbb{C}})$ it can be shown that $|C[N]| = |C|[|N|]$. So if $O = (C, N)$ is a subterm occurrence in $M \in \mathcal{T}(\tilde{\mathbb{C}})$, then $|O| = (|C|, |N|)$ is a subterm occurrence in $|M|$.

The reductor δ on $\mathcal{T}(\mathbb{C})$ gives rise to a δ-reductor $\tilde{\delta}$ on $\mathcal{T}(\tilde{\mathbb{C}})$.

5.2.7. DEFINITION. (i) δ_1 is the reductor on $\mathcal{T}(\tilde{\mathbb{C}})$ consisting of all reduction rules

$$\mathbf{C} M_1 \ldots M_m \to N$$

with $\delta : |\mathbf{C} M_1 \ldots M_m| \to N$.
(ii) δ_2 is the reductor on $\mathcal{T}(\tilde{\mathbb{C}})$ consisting of all reduction rules

$$\tilde{\mathbf{C}} M_1 \ldots M_m \to N$$

with $\delta : |\tilde{\mathbf{C}} M_1 \ldots M_m| \to N$.

(iii) $\tilde{\delta} = \delta_1\delta_2$. $\square$

Note that the contractum of a $\tilde{\delta}$-redex is an element of $\mathcal{T}(\mathbb{C})$, so in a $\tilde{\delta}$-contraction all markings disappear.

5.2.8. LEMMA (Projecting). *Let*

$$\pi' : M'_0 \to^{O'_1} \cdots \to^{O'_{n-1}} M'_n$$

be a $\beta\tilde{\delta}$-reduction sequence in $\mathcal{T}(\tilde{\mathbb{C}})$. Then the projection of π'

$$|\pi'| : |M'_0| \to^{|O'_1|} \cdots \to^{|O'_{n-1}|} |M'_n|$$

is a $\beta\delta$-reduction sequence in $\mathcal{T}(\mathbb{C})$.

PROOF. Simple. $\square$

5.2.9. LEMMA (Lifting). *Let π be a $\beta\delta$-reduction sequence in $\mathcal{T}(\mathbb{C})$ starting with M, and let $M' \in \mathcal{T}(\tilde{\mathbb{C}})$ be such that $|M'| = M$. Then there exists a unique $\beta\tilde{\delta}$-reduction sequence π' in $\mathcal{T}(\tilde{\mathbb{C}})$ starting with M' such that $|\pi'| = \pi$.*

PROOF. Although a formal proof of this lemma would be quite tedious, the truth of the Lifting Lemma is perfectly clear: one only needs to mark the right constant occurrences in π. $\square$

5.2.10. DEFINITION. Let $M, N \in \mathcal{T}(\mathbb{C})$ and let $\pi : M \twoheadrightarrow N$ be a $\beta\delta$-reduction sequence. Let $\mathcal{F}$ be a set of constant occurrences in M. Define $M' = (M, \mathcal{F}) \in \mathcal{T}(\tilde{\mathbb{C}})$. Let π' be the $\beta\tilde{\delta}$-reduction sequence starting with M' such that $|\pi'| = \pi$. Say $\pi' : M' \twoheadrightarrow N'$. Then $|N'| = N$, so there exists $\mathcal{G} \subseteq N$ with $N' = (N, \mathcal{G})$. Now define $\pi(\mathcal{F}) = \mathcal{G}$. The set $\pi(\mathcal{F})$ is a set of constant occurrences in N. The elements of $\pi(\mathcal{F})$ are called the residuals of $\mathcal{F}$ in N relative to π. $\square$

Using residuals of constant occurrences we will now define residuals of δ-redex occurrences.

Recall that we have a function $arity : |\mathbb{C}| \to \omega$ such that if $\mathbf{C}M_1 \ldots M_m$ is a δ-redex, then $m = arity(\mathbf{C})$.

5.2.11. DEFINITION. A quasi-δ-redex is a λ-term of the form $\mathbf{C}M_1 \ldots M_m$ with $m = arity(\mathbf{C})$. The constant $\mathbf{C}$ is called the head constant of $\mathbf{C}M_1 \ldots M_m$. $\square$

Note that every δ-redex is a quasi-δ-redex, but not vice versa. We will in fact define residuals of quasi-δ-redex occurrences, and not only residuals of δ-redex occurrences.

It can be shown (although we will not do this here) that if $\mathbf{C}$ is the head constant of a quasi-δ-redex occurrence in M, and $\pi : M \twoheadrightarrow N$ is a $\beta\delta$-reduction sequence, then every residual of $\mathbf{C}$ in N is the head constant of a quasi-δ-redex occurrence in N.

5.2.12. DEFINITION. Let $\pi : M \twoheadrightarrow N$ be a $\beta\delta$-reduction sequence in $\mathcal{T}(\mathbb{C})$ and let $\mathcal{F}$ be a set of quasi-δ-redex occurrences in M. Let $\mathcal{H}$ be the set of head constant occurrences of the quasi-δ-redex occurrences in $\mathcal{F}$. Then $\pi(\mathcal{H})$ is a set of constant occurrences in N, every one of which is the head constant of a quasi-δ-redex occurrence in N. Let $\pi(\mathcal{F})$ be the set of all quasi-δ-redex occurrences in N with a head constant occurrence in $\pi(\mathcal{H})$. The elements of $\pi(\mathcal{F})$ are called the residuals of $\mathcal{F}$ in N relative to π. $\square$

5.3. The Church-Rosser Property for $\lambda(\mathfrak{M})$.

Let $\mathfrak{M}$ be a type structure. We are going to show that $\lambda(\mathfrak{M}) \models \mathbf{CR}$ if and only if $\mathfrak{M}$ is quasi-extensional. Recall that $\lambda(\mathfrak{M}) = (\mathcal{T}(\mathfrak{M}), \beta\mu)$ and that μ is the least fixed point Φ^∞ of the monotone operator $\Phi : \mathbb{R} \to \mathbb{R}$. This least fixed point Φ^∞ is obtained by transfinite recursion: $\Phi^\zeta = \Phi(\Phi^{(\zeta)})$ with $\Phi^{(\zeta)} = \bigcup\{\Phi^\xi | \xi < \zeta\}$ for all $\zeta \in \mathbf{On}$; then $\Phi^\infty = \Phi^{|\Phi|}$ where $|\Phi|$ is the closure ordinal of Φ.

For every $\zeta \in \mathbf{On}$ the reductor Φ^ζ will be denoted by ζ, and the reductor $\Phi^{(\zeta)}$ will be denoted by (ζ). The notation ζ for Φ^ζ will not cause any confusion, as it will always be clear if we mean the ordinal ζ or the reductor ζ. Now we have reductors $\beta\zeta$ and $\beta(\zeta)$, and we have relations $\to_{\beta\zeta}$, $\to_{\beta(\zeta)}$, $\twoheadrightarrow_{\beta\zeta}$, $\twoheadrightarrow_{\beta(\zeta)}$, $=_{\beta\zeta}$, $=_{\beta(\zeta)}$, $\rhd_\zeta$ and $\rhd_{(\zeta)}$.

In the following we are going to prove $\lambda(\mathfrak{M}) \models \mathbf{CR}$ under the assumption that $\mathfrak{M}$ is quasi-extensional. We already proved the converse in section 5.1.

So suppose $\mathfrak{M}$ is quasi-extensional. By transfinite induction it will be shown that $\mathbf{CR}(\beta\zeta)$ for all $\zeta \in \mathbf{On}$. Taking $\zeta = |\Phi|$ then gives $\mathbf{CR}(\beta\mu)$.

Let $\zeta \in \mathbf{On}$. Assume, as the induction hypothesis, $\mathbf{CR}(\beta\xi)$ for all $\xi < \zeta$. Using the Second Church-Rosser Theorem it will be shown that $\mathbf{CR}(\beta\zeta)$. In the following we will prove that the reductor ζ is closed, deterministic, β-resistant and ζ-resistant.

5.3.1. LEMMA. *The reductor ζ is closed.*

PROOF. Every reductor of the form $\Phi(\mathbf{R})$ is closed. But ζ is $\Phi^\zeta = \Phi(\Phi^{(\zeta)})$ and therefore ζ is closed. $\square$

5.3.2. LEMMA. *For all $\sigma \in \mathbb{T}$, $M \in \mathcal{T}_\sigma(\mathfrak{M})^0$, $a, b \in \mathfrak{M}_\sigma$*

$$M \rhd_{(\zeta)} a \;\&\; M \rhd_{(\zeta)} b \Rightarrow a \approx b.$$

PROOF. Let $\sigma \in \mathbb{T}$, $M \in \mathcal{T}_\sigma(\mathfrak{M})^0$, $a, b \in \mathfrak{M}_\sigma$ be such that $M \rhd_{(\zeta)} a$ and $M \rhd_{(\zeta)} b$. Let $\vec{c} \in (\mathfrak{M})_\sigma$. Then

$$\mathbf{C}(a\vec{c}) =_{\beta(\zeta)} M\mathbf{C}(\vec{c}) =_{\beta(\zeta)} \mathbf{C}(b\vec{c})$$

so $\mathbf{C}(a\vec{c}) =_{\beta(\zeta)} \mathbf{C}(b\vec{c})$, so $\mathbf{C}(a\vec{c}) =_{\beta\xi} \mathbf{C}(b\vec{c})$ for some $\xi < \zeta$. But $\mathbf{CR}(\beta\xi)$, and $\mathbf{C}(a\vec{c})$ and $\mathbf{C}(b\vec{c})$ are $\beta\xi$-normal, so $\mathbf{C}(a\vec{c}) \equiv \mathbf{C}(b\vec{c})$ and so $a\vec{c} = b\vec{c}$. Therefore $a \approx b$. $\Box$

5.3.3. Corollary. *The reductor ζ is deterministic.*

Proof. Let $M \to N_1$ and $M \to N_2$ be ζ-rules. Then $M \equiv \mathbf{C}(a)M_1 \dots M_k$ and $N_1 \equiv \mathbf{C}(ab_1 \dots b_k)$ with $M_i \rhd_{(\zeta)} b_i$ $(i = 1, \dots, k)$, and $N_2 \equiv \mathbf{C}(ac_1 \dots c_k)$ with $M_i \rhd_{(\zeta)} c_i$ $(i = 1, \dots, k)$. By the previous lemma it follows that $\vec{b} \approx \vec{c}$. But $\mathfrak{M}$ is quasi-extensional, so a is extensional and so $a\vec{b} = a\vec{c}$. Therefore $N_1 \equiv N_2$, and ζ is deterministic. $\Box$

5.3.4. Lemma. *The reductor ζ is β-resistant.*

Proof. Let $\zeta : \mathbf{C}(a)M_1 \dots M_k \to \mathbf{C}(ab_1 \dots b_k)$ with $M_i \rhd_{(\zeta)} b_i$ $(i = 1, \dots, k)$. For every $i \in \{1, \dots, k\}$ let $N_i \in \mathcal{T}_{\sigma_i}(\mathfrak{M})^0$ with $M_i \twoheadrightarrow_\beta N_i$. Then $N_i \rhd_{(\zeta)} b_i$ $(i = 1, \dots, k)$, and so $\zeta : \mathbf{C}(a)N_1 \dots N_k \to \mathbf{C}(ab_1 \dots b_k)$. Therefore ζ is β-resistant. $\Box$

So far, so good. But we have saved the best for last: proving that ζ is ζ-resistant. Let us state the problem. Let

$$\zeta : \mathbf{C}(a)M_1 \dots M_k \to \mathbf{C}(ab_1 \dots b_k)$$

with $\sigma = \sigma_1 \to \dots \to \sigma_k \to 0 \in \mathbb{T}$, $\sigma \neq 0$, $a \in \mathfrak{M}_\sigma$, $(b_1, \dots, b_k) \in (\mathfrak{M})_\sigma$, $M_i \rhd_{(\zeta)} b_i$ $(i = 1, \dots, k)$. Let $j \in \{1, \dots, k\}$ and $M_j \to_\zeta N_j$. For $i \neq j$ define $N_i \equiv M_i$. If we can prove $\zeta : \mathbf{C}(a)N_1 \dots N_k \to \mathbf{C}(ab_1 \dots b_k)$ then ζ is ζ-resistant. This requires proving $N_j \rhd_{(\zeta)} b_j$. Let $\vec{c} \in (\mathfrak{M})_{\sigma_j}$. Then $M_j\mathbf{C}(\vec{c}) =_{\beta(\zeta)} \mathbf{C}(b_j\vec{c})$ and $M_j\mathbf{C}(\vec{c}) \to_\zeta N_j\mathbf{C}(\vec{c})$. We need $N_j\mathbf{C}(\vec{c}) =_{\beta(\zeta)} \mathbf{C}(b_j\vec{c})$. So we are done if we can prove that for all $P, Q \in \mathcal{T}_0(\mathfrak{M})^0$ and all $m \in \mathfrak{M}_0$

$$P \to_\zeta Q \;\&\; P =_{\beta(\zeta)} \mathbf{C}(m) \Rightarrow Q =_{\beta(\zeta)} \mathbf{C}(m).$$

We have this by the following lemma.

5.3.5. Lemma. *For all $\xi \in \mathbf{On}$ with $\xi \leq \zeta$, all $P, Q \in \mathcal{T}_0(\mathfrak{M})^0$ and all $m \in \mathfrak{M}_0$*

$$P \to_\zeta Q \;\&\; P =_{\beta(\xi)} \mathbf{C}(m) \Rightarrow Q =_{\beta(\xi)} \mathbf{C}(m). \;\Box$$

This lemma will be shown by transfinite induction on ξ. Because we now have a nested induction, we abbreviate the statement of this lemma. For every $\xi \in \mathbf{On}$ let $\mathbf{P}(\xi)$ be the statement: for all $P, Q \in \mathcal{T}_0(\mathfrak{M})^0$ and all $m \in \mathfrak{M}_0$

$$P \to_\zeta Q \;\&\; P =_{\beta(\xi)} \mathbf{C}(m) \Rightarrow Q =_{\beta(\xi)} \mathbf{C}(m).$$

The lemma then becomes $\forall\xi[\xi \leq \zeta \Rightarrow \mathbf{P}(\xi)]$. Note that we, almost trivially, have $\mathbf{P}(\xi)$ for $\xi > \zeta$. We left this case out of the lemma in order not to complicate the proof with an unnecessary case distinction.

PROOF. Let $\xi \in \mathbf{On}$. Assume as induction hypothesis

$$\forall\theta < \xi[\theta \leq \zeta \Rightarrow \mathbf{P}(\theta)].$$

Suppose $\xi \leq \zeta$. We are going to show $\mathbf{P}(\xi)$. Let $P, Q \in \mathcal{T}_0(\mathfrak{M})^0$ and $m \in \mathfrak{M}_0$ with $P \to_\zeta Q$ and $P =_{\beta(\xi)} \mathbf{C}(m)$. Say $P \to_\zeta^O Q$ with $O = (C, \Delta)$ and $\Delta \equiv \mathbf{C}(a)M_1 \dots M_k$ where $a \in \mathfrak{M}_\sigma$, $\sigma = \sigma_1 \to \dots \to \sigma_k \to 0 \in \mathbb{T}$, $\sigma \neq 0$, $\vec{b} = (b_1, \dots, b_k) \in (\mathfrak{M})_\sigma$, $M_i \in \mathcal{T}_{\sigma_i}(\mathfrak{M})^0$, $M_i \rhd_{(\zeta)} b_i$ $(i = 1, \dots, k)$ and $Q \equiv C[\mathbf{C}(a\vec{b})]$. Further we have $P =_{\beta\theta} \mathbf{C}(m)$ for some $\theta < \xi$. Then $\theta < \zeta$, so $\mathbf{CR}(\beta\theta)$ and so $P \twoheadrightarrow_{\beta\theta} \mathbf{C}(m)$. Let

$$\pi : P \equiv P_0 \to^{O_0} P_1 \to^{O_1} \cdots \to^{O_{n-1}} P_n \equiv \mathbf{C}(m)$$

be a $\beta\theta$-reduction sequence from P to $\mathbf{C}(m)$. For every $j < n$ let $O_j = (C_j, \Delta_j)$.

We are going to follow the ζ-redex occurrence O through the reduction sequence π and see what happens with its residuals.

For every $j \leq n$ let

$$\pi_j : P \equiv P_0 \to^{O_0} P_1 \to^{O_1} \cdots \to^{O_{j-1}} P_j$$

be the restriction of π to its first j steps, and define $\mathcal{F}_j = \pi_j(\{O\})$. Then $\mathcal{F}_j$ is the set of all residuals of O in P_j relative to π_j. (Note that $\mathcal{F}_0 = \{O\}$ and $\mathcal{F}_n = \emptyset$). Let $\mathcal{F}_j = \{O_1^j, O_2^j, \dots, O_{k_j}^j\}$ with $O_p^j = (C_p^j, \Delta_p^j)$ for $j = 1, 2, \dots, k_j$. About the residuals O_p^j we make the following remarks:

- All residuals O_p^j are closed, so nothing gets substituted into them.
- Let $j \leq n$ and $p \in \{1, \dots, k_j\}$. Then $\Delta \twoheadrightarrow_{\beta\theta} \Delta_p^j$ by internal reduction. So $\Delta_p^j \equiv \mathbf{C}(a)N_1 \dots N_k$ with $M_i \twoheadrightarrow_{\beta\theta} N_i$ $(i = 1, \dots, k)$. Because $M_i \rhd_{(\zeta)} b_i$ and $\theta < \zeta$ we have $N_i \rhd_{(\zeta)} b_i$. Therefore $\zeta : \Delta_p^j \to \mathbf{C}(a\vec{b})$ and Δ_p^j is also a ζ-redex.
- For every $j \in \{1, \dots, n\}$ the residuals $O_1^j, \dots, O_{k_j}^j$ are mutually disjoint, because none of the residuals of O can get substituted into another.

Assume that $O_1^j, \dots, O_{k_j}^j$ are such that if $1 \leq p < q \leq k_j$, then O_p^j is to the left of O_q^j in P_j.

Suppose for a moment, that at least one of the residuals of O is contracted somewhere in π. Then $O_j = O_p^j$ for some $j < n$ and $p \in \{1, \dots, k_j\}$. Then $\Delta_j \equiv \Delta_p^j \equiv \mathbf{C}(a)N_1 \dots N_k$ with $M_i \twoheadrightarrow_{\beta\theta} N_i$ $(i = 1, \dots, k_j)$. But Δ_j is a θ-redex, so $\theta : \Delta_j \to \mathbf{C}(a\vec{c})$ for some $\vec{c} = (c_1, \dots, c_k) \in (\mathfrak{M})_\sigma$ with $N_i \rhd_{(\theta)} c_i$ $(i = 1, \dots, k)$. We will prove $\vec{b} \approx \vec{c}$. Let $i \in \{1, \dots, k\}$ and $\vec{d} \in (\mathfrak{M})_{\sigma_i}$. Then

$$\mathbf{C}(b_i\vec{d}) =_{\beta(\zeta)} M_i\mathbf{C}(\vec{d}) \twoheadrightarrow_{\beta\theta} N_i\mathbf{C}(\vec{d}) =_{\beta(\theta)} \mathbf{C}(c_i\vec{d})$$

so $\mathbf{C}(b_i\vec{d}) =_{\beta\zeta_0} \mathbf{C}(c_i\vec{d})$ for some $\zeta_0 < \zeta$. But $\mathbf{CR}(\beta\zeta_0)$ and so $\mathbf{C}(b_i\vec{d}) \equiv \mathbf{C}(c_i\vec{d})$, so $b_i\vec{d} = c_i\vec{d}$. Therefore $b_i \approx c_i$ and $\vec{b} \approx \vec{c}$. Because a is extensional $a\vec{b} = a\vec{c}$, and we have

$$\Delta \twoheadrightarrow_{\beta\theta} \Delta_p^j \equiv \Delta_j \rightarrow_\theta \mathbf{C}(a\vec{c}) \equiv \mathbf{C}(a\vec{b})$$

so $\Delta \twoheadrightarrow_{\beta\theta} \mathbf{C}(a\vec{b})$ and therefore

$$P \equiv C[\Delta] \twoheadrightarrow_{\beta\theta} c[\mathbf{C}(a\vec{b})] \equiv Q$$

so $Q =_{\beta\theta} P =_{\beta(\xi)} \mathbf{C}(m)$. But $\theta < \xi$, so $Q =_{\beta(\xi)} \mathbf{C}(m)$.

So we may assume that no residual of O is ever contracted in the reduction sequence π. Then $O_j \neq O_p^j$ for all $j < n$ and all $p \in \{1, \ldots, k_j\}$. For every $j \leq n$ let Q_j be the result of contracting $O_1^j, \ldots, O_{k_j}^j$ in P_j, so that $P_j \twoheadrightarrow_\zeta Q_j$. Then $Q_0 \equiv Q$ and $Q_n \equiv \mathbf{C}(m)$.

Let $j < n$. We are going to prove that either $Q_j \equiv Q_{j+1}$ or $Q_j \rightarrow_{\beta\theta} Q_{j+1}$. In the following we consider two cases.

(i) $O_j \subseteq O_p^j$ for some $p \in \{1, \ldots, k_j\}$. Then $O_j \neq O_p^j$, so Δ_j is a proper subterm of Δ_p^j, and we have

$$P_j \equiv C_p^j[\Delta_p^j] \rightarrow_{\beta\theta}^{O_j} C_p^j[\Delta_p^{j+1}] \equiv P_{j+1}.$$

Therefore $Q_j \equiv Q_{j+1}$.

(ii) For all $p \in \{1, \ldots, k_j\}$ O_j is not contained in O_p^j. Then there are the following subcases:

(1) $P_j \rightarrow_\beta^{O_j} P_{j+1}$ and $\Delta_j \equiv (\lambda x.A)B$. Then we have the following situation:

$$\begin{tikzcd}
P_j \equiv C_j[(\lambda x.A)B] \arrow[r, two heads, "\zeta"] \arrow[d, "\beta"'] & Q_j \equiv C_j'[(\lambda x.A')B'] \\
P_{j+1} \equiv C_j[A[x := B]] \arrow[r, two heads, "\zeta"] & Q_{j+1} \equiv C_j'[A'[x := B']]
\end{tikzcd}$$

So, in this case $Q_j \rightarrow_\beta Q_{j+1}$.

(2) $P_j \rightarrow_\theta^{O_j} P_{j+1}$ and $\Delta_j \equiv \mathbf{C}(c)N_1 \ldots N_l$ with $c \in \mathfrak{M}_\tau$, $\tau = \tau_1 \rightarrow \ldots \rightarrow 0 \in \mathbb{T}$, $\tau \neq 0$, $N_i \in \mathcal{T}_{\tau_i}(\mathfrak{M})^0$, $N_i \rhd_{(\theta)} d_i$ $(i = 1, \ldots, l)$ for some $\vec{d} = (d_1, \ldots, d_l) \in (\mathfrak{M})_\tau$. Now we have the following situation:

$$\begin{array}{ccc} P_j \equiv C_j[\mathbf{C}(c)\vec{N}] & \xrightarrow{\zeta} & Q_j \equiv C'_j[\mathbf{C}(c)\vec{N}'] \\ \Big\downarrow \theta & & \\ P_{j+1} \equiv C_j[\mathbf{C}(c\vec{d})] & \xrightarrow{\zeta} & Q_{j+1} \equiv C'_j[\mathbf{C}(c\vec{d})] \end{array}$$

with $N_i \twoheadrightarrow_\zeta N'_i$ $(i = 1, \ldots, l)$. Let $i \in \{1, \ldots, l\}$ and

$$N_i \equiv N_i^0 \rightarrow_\zeta N_i^1 \rightarrow_\zeta \cdots \rightarrow_\zeta N_i^r \equiv N'_i.$$

We are going to prove that for all $s \in \omega$

$$s \leq r \Rightarrow N_i^s \rhd_{(\theta)} d_i.$$

This will be done by induction on s.

- $s = 0$: $N_i^0 \equiv N_i \rhd_{(\theta)} d_i$.
- Let $s + 1 \leq r$. Then $s \leq r$ and so, by induction $N_i^s \rhd_{(\theta)} d_i$. Let $\vec{e} \in (\mathfrak{M})_{\tau_i}$. Then $N_i^s\mathbf{C}(\vec{e}) =_{\beta(\theta)} \mathbf{C}(d_i\vec{e})$ and $N_i^s\mathbf{C}(\vec{e}) \rightarrow_\zeta N_i^{s+1}\mathbf{C}(\vec{e})$. But $\mathbf{P}(\theta)$, so $N_i^{s+1}\mathbf{C}(\vec{e}) =_{\beta(\theta)} \mathbf{C}(d_i\vec{e})$. Therefore $N_i^{s+1} \rhd_{(\theta)} d_i$.

It follows that $N'_i \rhd_{(\theta)} d_i$ $(i = 1, \ldots, l)$. So $\theta : \mathbf{C}(c)\vec{N}' \to \mathbf{C}(c\vec{d})$, and therefore $Q_j \rightarrow_\theta Q_{j+1}$.

We see that in all of the cases we have either $Q_j \equiv Q_{j+1}$ or $Q_j \rightarrow_{\beta\theta} Q_{j+1}$. Therefore $Q_0 \twoheadrightarrow_{\beta\theta} Q_n$ and so $Q \twoheadrightarrow_{\beta\theta} \mathbf{C}(m)$. So $Q =_{\beta(\xi)} \mathbf{C}(m)$, because $\theta < \xi$.

Now we have $\mathbf{P}(\xi)$. By transfinite induction it follows that

$$\forall\xi[\xi \leq \zeta \Rightarrow \mathbf{P}(\xi)]$$

and we are done. □

5.3.6. Corollary. *The reductor ζ is ζ-resistant.*

Proof. It suffices to show that ζ can resist internal one-step ζ-reductions. Let

$$\zeta : \mathbf{C}(a)M_1 \ldots M_k \to \mathbf{C}(ab_1 \ldots b_k)$$

with $\sigma = \sigma_1 \rightarrow \cdots \rightarrow \sigma_k \rightarrow 0 \in \mathbb{T}$, $\sigma \neq 0$, $a \in \mathfrak{M}_\sigma$, $(b_1, \ldots, b_k) \in (\mathfrak{M})_\sigma$, $M_i \rhd_{(\zeta)} b_i$, $(i = 1, \ldots, k)$. Let $i \in \{1, \ldots, k\}$ and $M_i \rightarrow_\zeta N_i$. For $j \neq i$ define $N_j \equiv M_j$. We need to prove $\zeta : \mathbf{C}(a)\vec{N} \to \mathbf{C}(a\vec{b})$. Let $\vec{c} \in (\mathfrak{M})_{\sigma_i}$. Then $M_i\mathbf{C}(\vec{c}) =_{\beta(\zeta)} \mathbf{C}(b_i\vec{c})$ and $M_i\mathbf{C}(\vec{c}) \rightarrow_\zeta N_i\mathbf{C}(\vec{c})$. By the previous lemma it follows that $N_i\mathbf{C}(\vec{c}) =_{\beta(\zeta)} \mathbf{C}(b_i\vec{c})$. Therefore $N_i \rhd_{(\zeta)} b_i$. Also $N_j \rhd_{(\zeta)} b_j$ for $j \neq i$. So indeed $\zeta : \mathbf{C}(a)\vec{N} \to \mathbf{C}(a\vec{b})$, and therefore ζ is ζ-resistant. □

Altogether we have now shown that the reductor ζ is closed, deterministic, β-resistant and ζ-resistant, and therefore also $\beta\zeta$-resistant. Using the Second Church-Rosser Theorem it follows that $\mathbf{CR}(\beta\zeta)$. By transfinite induction we obtain the following theorem.

5.3.7. THEOREM. *Let $\mathfrak{M}$ be quasi-extensional. Then $\mathbf{CR}(\beta\zeta)$ for all $\zeta \in \mathbf{On}$. In particular $\mathbf{CR}(\beta\mu)$.* $\square$

5.3.8. THEOREM (Church-Rosser Theorem for diagrams).
$\lambda(\mathfrak{M}) \models \mathbf{CR}$ *if and only if $\mathfrak{M}$ is quasi-extensional.* $\square$

5.3.9. COROLLARY. *Let $\mathfrak{M}$ be quasi-extensional. Then for all $\sigma \in \mathbb{T}$, $M \in \mathcal{T}_\sigma(\mathfrak{M})^0$, $a, b \in \mathfrak{M}_\sigma$*

$$M \rhd a \,\&\, M \rhd b \Rightarrow a \approx b.$$

PROOF. Let $\sigma \in \mathbb{T}$, $M \in \mathcal{T}_\sigma(\mathfrak{M})^0$, $a, b \in \mathfrak{M}_\sigma$ with $M \rhd a$ and $M \rhd b$. Let $\vec{c} \in (\mathfrak{M})_\sigma$. Then

$$\mathbf{C}(a\vec{c}) =_{\beta\mu} M\mathbf{C}(\vec{c}) =_{\beta\mu} \mathbf{C}(b\vec{c}).$$

But $\mathbf{CR}(\beta\mu)$, and $\mathbf{C}(a\vec{c})$ and $\mathbf{C}(b\vec{c})$ are $\beta\mu$-normal forms, so $\mathbf{C}(a\vec{c}) \equiv \mathbf{C}(b\vec{c})$ and so $a\vec{c} = b\vec{c}$. Therefore $a \approx b$. $\square$

5.4. The compatibility of $\lambda(\mathfrak{M})$-computability.

The relation $\rhd$ of $\lambda(\mathfrak{M})$-computability is a relation between $\mathcal{T}(\mathfrak{M})^0$ and $|\mathfrak{M}|$. In this section it will be shown that if $\mathfrak{M}$ is quasi-extensional, then $\rhd$ is compatible with application in $\mathcal{T}(\mathfrak{M})^0$ and in $|\mathfrak{M}|$.

5.4.1. THEOREM (Compatibility Theorem).
Let $\mathfrak{M}$ be quasi-extensional. Then for all $\sigma, \tau \in \mathbb{T}$, $M \in \mathcal{T}_{\sigma\to\tau}(\mathfrak{M})^0$, $N \in \mathcal{T}_\sigma(\mathfrak{M})^0$, $a \in \mathfrak{M}_{\sigma\to\tau}$, $b \in \mathfrak{M}_\sigma$

$$M \rhd a \,\&\, N \rhd b \Rightarrow MN \rhd ab. \square$$

We will prove the Compatibility Theorem by induction on the structure of σ. In the following $\mathfrak{M}$ is a quasi-extensional type structure.

5.4.2. LEMMA (Induction Basis).
For all $\tau \in \mathbb{T}$, $M \in \mathcal{T}_{0\to\tau}(\mathfrak{M})^0$, $N \in \mathcal{T}_0(\mathfrak{M})^0$, $a \in \mathfrak{M}_{0\to\tau}$, $b \in \mathfrak{M}_0$

$$M \rhd a \,\&\, N \rhd b \Rightarrow MN \rhd ab.$$

PROOF. Let $\tau \in \mathbb{T}$, $M \in \mathcal{T}_{0\to\tau}(\mathfrak{M})^0$, $N \in \mathcal{T}_0(\mathfrak{M})^0$, $a \in \mathfrak{M}_{0\to\tau}$, $b \in \mathfrak{M}_0$ with $M \rhd a$ and $N \rhd b$. Then $N =_{\beta\mu} \mathbf{C}(b)$, so for all $\vec{c} \in (\mathfrak{M})_\tau$

$$MN\mathbf{C}(\vec{c}) =_{\beta\mu} M\mathbf{C}(b)\mathbf{C}(\vec{c}) =_{\beta\mu} \mathbf{C}(ab\vec{c}).$$

So $MN \rhd ab$. $\square$

For every $\sigma \in \mathbb{T}$ let $\mathbf{P}(\sigma)$ be the statement: for all $\tau \in \mathbb{T}$, $M \in \mathcal{T}_{\sigma\to\tau}(\mathfrak{M})^0$, $N \in \mathcal{T}_\sigma(\mathfrak{M})^0$, $a \in \mathfrak{M}_{\sigma\to\tau}$, $b \in \mathfrak{M}_\sigma$

$$M \rhd a \,\&\, N \rhd b \Rightarrow MN \rhd ab.$$

By the previous lemma we already have $\mathbf{P}(0)$. Now let $\sigma = \sigma_1 \to \cdots \to \sigma_k \to 0$, $\sigma \neq 0$ and suppose, as the induction hypothesis, that $\mathbf{P}(\sigma_1), \ldots, \mathbf{P}(\sigma_k)$ hold. We will prove $\mathbf{P}(\sigma)$. Let $\tau \in \mathbb{T}$, $M \in \mathcal{T}_{\sigma\to\tau}(\mathfrak{M})^0$, $N \in \mathcal{T}_\sigma(\mathfrak{M})^0$, $a \in \mathfrak{M}_{\sigma\to\tau}$, $b \in \mathfrak{M}_\sigma$ be such that $M \rhd a$ and $N \rhd b$. We have to show that $MN \rhd ab$. Let $\vec{c} \in (\mathfrak{M})_\tau$. Then $M\mathbf{C}(b)\mathbf{C}(\vec{c}) =_{\beta\mu} \mathbf{C}(ab\vec{c})$, because $M \rhd a$. Because $\mathfrak{M}$ is quasi-extensional we have $\mathbf{CR}(\beta\mu)$, and so $M\mathbf{C}(b)\mathbf{C}(\vec{c}) \twoheadrightarrow_{\beta\mu}\mathbf{C}(ab\vec{c})$. The idea is now to replace $\mathbf{C}(b)$ by N in this reduction, and to prove that the reduction thus obtained still holds. This is the idea behind the following lemma. This lemma shows that if $L \twoheadrightarrow_{\beta\mu}\mathbf{C}(m)$, then we can pick a number of occurrences of $\mathbf{C}(b)$ in L and replace them by N, without affecting the reduction. In this lemma and its proof we use marked constants. The marked version of $\mathbf{C}(b)$ is denoted by $\tilde{\mathbf{C}}(b)$. Recall that if $\mathcal{F}$ is a set of occurrences of $\mathbf{C}(b)$ in a term L, then $(L, \mathcal{F})$ denotes the result of marking all occurrences of $\mathbf{C}(b)$ in L that are in $\mathcal{F}$.

5.4.3. Lemma. *Let $\zeta \in \mathbf{On}$. Let $L \in \mathcal{T}_0(\mathfrak{M})^0$, let $\mathcal{F}$ be a set of occurrences of $\mathbf{C}(b)$ in L, and let $m \in \mathfrak{M}_0$. Then*

$$L \twoheadrightarrow_{\beta\zeta}\mathbf{C}(m) \Rightarrow (L, \mathcal{F})[\tilde{\mathbf{C}}(b) := N] \twoheadrightarrow_{\beta\mu}\mathbf{C}(m).$$

Proof. For every $\zeta \in \mathbf{On}$ let $\mathbf{Q}(\zeta)$ be the statement: for every $L \in \mathcal{T}_0(\mathfrak{M})^0$, every set $\mathcal{F}$ of occurrences of $\mathbf{C}(b)$ in L and every $m \in \mathfrak{M}_0$

$$L \twoheadrightarrow_{\beta\zeta}\mathbf{C}(m) \Rightarrow (L, \mathcal{F})[\tilde{\mathbf{C}}(b) := N] \twoheadrightarrow_{\beta\mu}\mathbf{C}(m).$$

By transfinite induction it will be shown that $\forall\zeta \in \mathbf{On}[\mathbf{Q}(\zeta)]$. Let $\zeta \in \mathbf{On}$ and assume $\mathbf{Q}(\xi)$ for all $\xi < \zeta$. We will prove $\mathbf{Q}(\zeta)$. Let $L \in \mathcal{T}_0(\mathfrak{M})^0$, let $\mathcal{F}$ be a set of occurrences of $\mathbf{C}(b)$ in L, and let $m \in \mathfrak{M}_0$. Suppose $L \twoheadrightarrow_{\beta\zeta}\mathbf{C}(m)$. Let

$$\pi : L \equiv L_0 \to^{O_0} L_1 \to^{O_1} \cdots \to^{O_{n-1}} L_n \equiv \mathbf{C}(m)$$

be a $\beta\zeta$-reduction sequence. For each $i \leq n$ let π_i be the restriction of π to its first i steps, so

$$\pi_i : L_0 \to^{O_0} L_1 \to^{O_1} \cdots \to^{O_{i-1}} L_i.$$

For every $i \leq n$ define $\mathcal{F}_i = \pi_i(\mathcal{F})$. Then $\mathcal{F}_i$ is the set of all residuals of $\mathcal{F}$ in L_i relative to π_i. For every $i \leq n$ and every subterm occurrence P in L_i let P^* be obtained from P by replacing every occurrence of $\mathbf{C}(b)$ from $\mathcal{F}_i$ that is in P by N. Note that we confuse subterms and subterm occurrences. This is done not to unnecessarily complicate matters. Also note that

$$L_i^* \equiv (L_i, \mathcal{F}_i)[\tilde{\mathbf{C}}(b) := N].$$

Let $i < n$. It will be shown that $L_i^* \twoheadrightarrow_{\beta\mu} L_{i+1}^*$. Distinguish the following two cases.

(i) $L_i \to_{\beta}^{O_i} L_{i+1}$. Then by lifting we have $(L_i, \mathcal{F}_i) \to_\beta (L_{i+1}, \mathcal{F}_{i+1})$ and therefore

$$L_i^* \equiv (L_i, \mathcal{F}_i)[\tilde{\mathbf{C}}(b) := N] \to_\beta (L_{i+1}, \mathcal{F}_{i+1})[\tilde{\mathbf{C}}(b) := N] \equiv L_{i+1}^*.$$

(ii) $L_i \to_{\zeta}^{O_i} L_{i+1}$. Let $O_i = (C_i, \Delta_i)$ with $\Delta_i \equiv \mathbf{C}(d) N_1 \dots N_p$, $v = v_1 \to \dots \to v_p \to 0 \in \mathbb{T}$, $v \neq 0$, $d \in \mathfrak{M}_v$, $\vec{e} = (e_1, \dots, e_p) \in (\mathfrak{M})_v$, $N_j \in \mathcal{T}_{v_j}(\mathfrak{M})^0$, $N_j \rhd_{(\zeta)} e_j$ $(j = 1, \dots, p)$.

First it will be shown that for every $j \in \{1, \dots, p\}$ we have $N_j^* \rhd e_j$. Let $j \in \{1, \dots, p\}$ and let $\vec{f} \in (\mathfrak{M})_{v_j}$. Then $N_j \mathbf{C}(\vec{f}) =_{\beta\xi} \mathbf{C}(e_j \vec{f})$ for some $\xi < \zeta$. But $\mathbf{CR}(\beta\xi)$, so $N_j \mathbf{C}(\vec{f}) \twoheadrightarrow_{\beta\xi} \mathbf{C}(e_j \vec{f})$. By $\mathbf{Q}(\xi)$ it follows that $N_j^* \mathbf{C}(\vec{f}) \twoheadrightarrow_{\beta\mu} \mathbf{C}(e_j \vec{f})$. Therefore $N_j^* \rhd e_j$.

Now there are two subcases.

(1) The occurrence of $\mathbf{C}(d)$ in L_i is not an element of $\mathcal{F}_i$. Then we have $\Delta_i^* \equiv \mathbf{C}(d) N_1^* \dots N_p^*$, so $\Delta_i^* \to_\mu \mathbf{C}(d e_1 \dots e_p)$ and so $L_i^* \to_\mu L_{i+1}^*$.

(2) The occurrence of $\mathbf{C}(d)$ in L_i is an element of $\mathcal{F}_i$. Then $d = b$, $v = \sigma$, $p = k$ and $v_j = \sigma_j$ $(j = 1, \dots, p = k)$. And also

$$\Delta_i^* \equiv N N_1^* \dots N_p^*.$$

Recall that we have $N \rhd b = d$ and $N_j^* \rhd e_j$ $(j = 1, \dots, p)$. Now by successive application of $\mathbf{P}(\sigma_1)$,...,$\mathbf{P}(\sigma_k)$ we obtain

$$N N_1^* N_2^* \dots N_j^* \rhd d e_1 \dots e_j \ \ (j = 1, \dots, p).$$

Taking $j = p$ gives $N N_1^* \dots N_p^* \rhd d e_1 \dots e_p$, so $N N_1^* \dots N_p^* =_{\beta\mu} \mathbf{C}(d e_1 \dots e_p)$. Because $\mathbf{CR}(\beta\mu)$ this gives

$$N N_1^* \dots N_p^* \twoheadrightarrow_{\beta\mu} \mathbf{C}(d e_1 \dots e_p)$$

and so $L_i^* \twoheadrightarrow_{\beta\mu} L_{i+1}^*$.

In any case $L_i^* \twoheadrightarrow_{\beta\mu} L_{i+1}^*$. So we have $L_i^* \twoheadrightarrow_{\beta\mu} L_{i+1}^*$ for every $i < n$. Therefore $L_0^* \twoheadrightarrow_{\beta\mu} L_n^*$, and so

$$(L, \mathcal{F})[\tilde{\mathbf{C}}(b) := N] \twoheadrightarrow_{\beta\mu} \mathbf{C}(m).$$

Therefore $\mathbf{Q}(\zeta)$. By transfinite induction the lemma now follows. □

5.4.4. Corollary. *Let $L \in \mathcal{T}_0(\mathfrak{M})^0$, let $\mathcal{F}$ be a set of occurrences of $\mathbf{C}(b)$ in L and let $m \in \mathfrak{M}_0$. Then*

$$L \twoheadrightarrow_{\beta\mu} \mathbf{C}(m) \Rightarrow (L, \mathcal{F})[\tilde{\mathbf{C}}(b) := N] \twoheadrightarrow_{\beta\mu} \mathbf{C}(m). \ \square$$

Now we can finish the proof of $\mathbf{P}(\sigma)$. Recall that $M \mathbf{C}(b) \mathbf{C}(\vec{c}) \twoheadrightarrow_{\beta\mu} \mathbf{C}(a b \vec{c})$. Let O be the occurrence $(M \square^\sigma \mathbf{C}(\vec{c}), \mathbf{C}(b))$ of $\mathbf{C}(b)$ in $M \mathbf{C}(b) \mathbf{C}(\vec{c})$, and let $\mathcal{F} = \{O\}$. By the previous corollary we have

$$M N \mathbf{C}(\vec{c}) \equiv (M \mathbf{C}(b) \mathbf{C}(\vec{c}), \mathcal{F})[\tilde{\mathbf{C}}(b) := N] \twoheadrightarrow_{\beta\mu} \mathbf{C}(a b \vec{c}).$$

Therefore $MN \rhd ab$ and $\mathbf{P}(\sigma)$. By induction $\forall \sigma \in \mathbb{T}[\mathbf{P}(\sigma)]$. This finishes the proof of the Compatibility Theorem.

5.5. Diagrams of λ-structures.

Let $\mathfrak{M}$ be a λ-structure. There exists a quasi-$\lambda(\mathfrak{M})$-interpretation $[\![\,]\!]$ in $\mathfrak{M}$ such that $[\![\mathbf{C}(a)]\!] = a$ for all $a \in |\mathfrak{M}|$. It will be shown that μ-reduction and $\lambda(\mathfrak{M})$-computability can be characterized in terms of this quasi-interpretation in a very simple way. Moreover, with this quasi-interpretation $\mathfrak{M}$ becomes a quasi-$\lambda(\mathfrak{M})$-model.

Because $\mathfrak{M}$ is quasi-extensional, we have:

- $\lambda(\mathfrak{M}) \models \mathbf{CR}$;
- the relation $\,\triangleright\,$ is compatible;
- $M \,\triangleright a \,\&\, N \,\triangleright b \Rightarrow a \approx b$.

5.5.1. THEOREM. *Let $M \in \mathcal{T}(\mathfrak{M})$. Then for all $\sigma = \sigma_1 \rightarrow \cdots \rightarrow \sigma_k \rightarrow 0 \in \mathbb{T}$, all $\vec{x} = x_1, \ldots, x_k$ with $x_i \in Var_{\sigma_i}$, all $\vec{a} = (a_1, \ldots, a_k) \in (\mathfrak{M})_\sigma$ and all $\rho \in Val(\mathfrak{M})$: if $FV(M) \subseteq \{\vec{x}\}$, then*

$$M[\vec{x} := \mathbf{C}(\vec{a})] \,\triangleright [\![M]\!]_{\rho(\vec{x}:=\vec{a})}.$$

PROOF. By induction on the structure of M.

(i) $M \equiv x \equiv x_i$:

$$x[\vec{x} := \mathbf{C}(\vec{a})] \equiv \mathbf{C}(a_i) \,\triangleright a_i = [\![x]\!]_{\rho(\vec{x}:=\vec{a})}.$$

(ii) $M \equiv \mathbf{C}(b)$:

$$\mathbf{C}(b)[\vec{x} := \mathbf{C}(\vec{a})] \equiv \mathbf{C}(b) \,\triangleright b = [\![\mathbf{C}(b)]\!]_{\rho(\vec{x}:=\vec{a})}.$$

(iii) $M \equiv M_1M_2$. Let $FV(M) \subseteq \{\vec{x}\}$. Then $FV(M_1) \subseteq \{\vec{x}\}$ and $FV(M_2) \subseteq \{\vec{x}\}$, so by induction

$$M_1[\vec{x} := \mathbf{C}(\vec{a})] \,\triangleright [\![M_1]\!]_{\rho(\vec{x}:=\vec{a})}$$
$$M_2[\vec{x} := \mathbf{C}(\vec{a})] \,\triangleright [\![M_2]\!]_{\rho(\vec{x}:=\vec{a})}.$$

Using the Compatibility Theorem it follows that

$$(M_1M_2)[\vec{x} := \mathbf{C}(\vec{a})] \,\triangleright [\![M_1M_2]\!]_{\rho(\vec{x}:=\vec{a})}.$$

(iv) $M \equiv \lambda x.M_1$ with $x \in Var_\upsilon$ and $M_1 \in \mathcal{T}_\tau(\mathfrak{M})$. Let $FV(M) \subseteq \{x_1, \ldots, x_k\}$. Then $FV(M_1) \subseteq \{x_1, \ldots, x_k, x\}$ so by induction it follows with a simple calculation that for all $a \in \mathfrak{M}_\upsilon$ and $\vec{b} \in (\mathfrak{M})_\tau$ we have

$$(\lambda x.M_1)[\vec{x} := \mathbf{C}(\vec{a})]\mathbf{C}(a)\mathbf{C}(\vec{b}) =_{\beta\mu} \mathbf{C}([\![\lambda x.M_1]\!]_{\rho(\vec{x}:=\vec{a})}a\vec{b}).$$

Therefore $(\lambda x.M_1)[\vec{x} := \mathbf{C}(\vec{a})] \,\triangleright [\![\lambda x.M_1]\!]_{\rho(\vec{x}:=\vec{a})}$. $\square$

5.5.2. Corollary. *For every $M \in \mathcal{T}(\mathfrak{M})^0$ we have $M \rhd [\![M]\!]$.* $\Box$

From this corollary we obtain the following characterization of $\lambda(\mathfrak{M})$-computability.

5.5.3. Theorem (First Characterization Theorem).
For all $\sigma \in \mathbb{T}$, $M \in \mathcal{T}_\sigma(\mathfrak{M})^0$, $a \in \mathfrak{M}_\sigma$

$$M \rhd a \Longleftrightarrow [\![M]\!] \approx a.$$

Proof. Let $\sigma \in \mathbb{T}$, $M \in \mathcal{T}_\sigma(\mathfrak{M})^0$, $a \in \mathfrak{M}_\sigma$.
(i) Suppose $M \rhd a$. By the corollary $M \rhd [\![M]\!]$, and therefore $[\![M]\!] \approx a$.
(ii) Suppose $[\![M]\!] \approx a$. By the corollary $M \rhd [\![M]\!]$, and so $M \rhd a$. $\Box$

Using the First Characterization Theorem we can also characterize μ-reduction in terms of $[\![\;]\!]$.

5.5.4. Definition. The reductor ι on $\mathcal{T}(\mathfrak{M})$ consists of all reduction rules

$$\mathbf{C}(a)M_1 \ldots M_k \to \mathbf{C}(a[\![M_1]\!] \ldots [\![M_k]\!])$$

with $\sigma = \sigma_1 \to \cdots \to \sigma_k \to 0 \in \mathbb{T}$, $\sigma \neq 0$, $a \in \mathfrak{M}_\sigma$ and $M_i \in \mathcal{T}_{\sigma_i}(\mathfrak{M})^0$ $(i = 1, \ldots, k)$.
$\Box$

5.5.5. Theorem. $\mu = \iota$.

Proof. We prove $\mu \subseteq \iota$ and $\iota \subseteq \mu$.
(i) Let $(M, N) \in \mu$. Then $M \equiv \mathbf{C}(a)M_1 \ldots M_k$ and $N \equiv \mathbf{C}(ab_1 \ldots b_k)$ with $M_i \rhd b_i$ $(i = 1, \ldots, k)$. By the First Characterization Theorem $[\![M_i]\!] \approx b_i$ $(i = 1, \ldots, k)$. Because a is extensional this gives

$$ab_1 \ldots b_k = a[\![M_1]\!] \ldots [\![M_k]\!]$$

so $N \equiv \mathbf{C}(a[\![M_1]\!] \ldots [\![M_k]\!])$. Therefore $(M, N) \in \iota$ and $\mu \subseteq \iota$.
(ii) Let $(M, N) \in \iota$. Then $M \equiv \mathbf{C}(a)M_1 \ldots M_k$ and $N \equiv \mathbf{C}(a[\![M_1]\!] \ldots [\![M_k]\!])$. Again by the First Characterization Theorem $M_i \rhd [\![M_i]\!]$ $(i = 1, \ldots, k)$ and therefore $(M, N) \in \mu$. So $\iota \subseteq \mu$. $\Box$

5.5.6. Theorem (Second Characterization Theorem).
The reductor μ on $\mathcal{T}(\mathfrak{M})$ consists of all reduction rules

$$\mathbf{C}(a)M_1 \ldots M_k \to \mathbf{C}(a[\![M_1]\!] \ldots [\![M_k]\!])$$

with $\sigma = \sigma_1 \to \cdots \to \sigma_k \to 0 \in \mathbb{T}$, $\sigma \neq 0$, $a \in \mathfrak{M}_\sigma$ and $M_i \in \mathcal{T}_{\sigma_i}(\mathfrak{M})^0$ $(i = 1, \ldots, k)$.
$\Box$

5.5.7. THEOREM. *$(\mathfrak{M}, [\![\;]\!])$ is a quasi-$\lambda(\mathfrak{M})$-model.*

PROOF. Because $\mathfrak{M}$ is quasi-extensional, we only need to prove quasi-satisfaction of the μ-axioms in $(\mathfrak{M}, [\![\;]\!])$. Let $(M, N) \in \mu$. Then $M \equiv \mathbf{C}(a[\![M_1]\!] \dots [\![M_k]\!])$ and $N \equiv \mathbf{C}(a[\![M_1]\!] \dots [\![M_k]\!])$ by the Second Characterization Theorem. Now compute

$$\begin{aligned} [\![M]\!] &= [\![\mathbf{C}(a)M_1 \dots M_k]\!] \\ &= [\![\mathbf{C}(a)]\!][\![M_1]\!] \dots [\![M_k]\!] \\ &= a[\![M_1]\!] \dots [\![M_k]\!] \\ &= [\![\mathbf{C}(a[\![M_1]\!] \dots [\![M_k]\!])]\!] \\ &= [\![N]\!]. \end{aligned}$$

Therefore $\mathfrak{M}, [\![\;]\!] \models_q M = N$ (even $\mathfrak{M}, [\![\;]\!] \models M = N$). Using theorem 2.2.14 it follows that $(\mathfrak{M}, [\![\;]\!])$ is a quasi-$\lambda(\mathfrak{M})$-model. □

5.6. Type-free diagrams.

The diagram construction in section 5.1 can also be carried out in the type-free λ-calculus. Then every type structure $\mathfrak{M}$ gives rise to a type-free λ-calculus $\mathcal{D}(\mathfrak{M})$ called the type-free diagram of $\mathfrak{M}$. Below we outline the construction of $\mathcal{D}(\mathfrak{M})$.

5.6.1. DEFINITION. The set $\Lambda(\mathfrak{M})$ is defined inductively as follows.
(i) $Var = \{v_0, v_1, v_2, \dots\} \subseteq \Lambda(\mathfrak{M})$
(ii) $a \in |\mathfrak{M}| \Rightarrow \mathbf{C}(a) \in \Lambda(\mathfrak{M})$
(iii) $M, N \in \Lambda(\mathfrak{M}) \Rightarrow (MN) \in \Lambda(\mathfrak{M})$
(iv) $x \in Var, M \in \Lambda(\mathfrak{M}) \Rightarrow (\lambda x.M) \in \Lambda(\mathfrak{M})$. □

Adapting the construction in section 5.1 we obtain a reductor μ on $\Lambda(\mathfrak{M})$ consisting of reduction rules

$$\mathbf{C}(a)M_1 \dots M_k \to \mathbf{C}(ab_1 \dots b_k)$$

with $\sigma = \sigma_1 \to \dots \to \sigma_k \to 0 \neq 0$, $a \in \mathfrak{M}_\sigma$, $(b_1, \dots, b_k) \in (\mathfrak{M})_\sigma$, $M_1, \dots, M_k \in \Lambda(\mathfrak{M})^0$ and $M_i \rhd b_i$ $(i = 1, \dots, k)$, where the relation $\rhd$ is given as follows. For all $M \in \Lambda(\mathfrak{M})^0$, $\sigma \in \mathbb{T}$, $a \in \mathfrak{M}_\sigma$ we have $M \rhd a$ if and only if for all $(b_1, \dots, b_k) \in (\mathfrak{M})_\sigma$

$$M\mathbf{C}(b_1) \dots \mathbf{C}(b_k) =_{\beta\mu} \mathbf{C}(ab_1 \dots b_k).$$

Now the type-free diagram of $\mathfrak{M}$ is defined by

$$\mathcal{D}(\mathfrak{M}) = (\Lambda(\mathfrak{M}), \beta\mu).$$

Using essentially the same methods as in sections 5.3 and 5.4 the following two theorems can be shown.

5.6.2. THEOREM (Church-Rosser Theorem for type-free diagrams).
$\mathcal{D}(\mathfrak{M}) \models \mathbf{CR}$ *if and only if* $\mathfrak{M}$ *is quasi-extensional.* □

5.6.3. THEOREM (Compatibility Theorem for type-free diagrams).
If $\mathfrak{M}$ *is quasi-extensional, then for all* $M, N \in \Lambda(\mathfrak{M})^0$, $\sigma, \tau \in \mathbb{T}$, $a \in \mathfrak{M}_{\sigma\to\tau}$, $b \in \mathfrak{M}_\sigma$

$$M \ \triangleright a \,\&\, N \ \triangleright b \Rightarrow MN \ \triangleright ab. \square$$

Let $\mathfrak{M}$ be such that $\mathfrak{M}_0 = \omega$. Then there exists a useful variant of $\mathcal{D}(\mathfrak{M})$ called the ω-diagram of $\mathfrak{M}$ which is denoted by $\mathcal{D}_\omega(\mathfrak{M})$.

5.6.4. DEFINITION. (i) The set $\Lambda_\omega(\mathfrak{M})$ is defined inductively as follows.

(1) $Var \subseteq \Lambda_\omega(\mathfrak{M})$

(2) $a \in \mathfrak{M}_\sigma \,\&\, \sigma \neq 0 \Rightarrow \mathbf{C}(a) \in \Lambda_\omega(\mathfrak{M})$

(3) $M, N \in \Lambda_\omega(\mathfrak{M}) \Rightarrow (MN) \in \Lambda_\omega(\mathfrak{M})$

(4) $x \in Var, M \in \Lambda_\omega(\mathfrak{M}) \Rightarrow (\lambda x.M) \in \Lambda_\omega(\mathfrak{M})$.

(ii) For every $m \in \omega$ define $\mathbf{C}(m) \equiv \ulcorner m \urcorner$. □

Recall that $\ulcorner 0 \urcorner, \ulcorner 1 \urcorner, \ulcorner 2 \urcorner, \ldots$ is the sequence of numerals in the pure type-free λ-calculus defined in section 4.1.

On $\Lambda_\omega(\mathfrak{M})$ we have a reductor μ_ω consisting of all reduction rules

$$\mathbf{C}(a)M_1 \ldots M_k \to \mathbf{C}(ab_1 \ldots b_k)$$

with $M_1, \ldots, M_k \in \Lambda_\omega(\mathfrak{M})^0$ and $M_i \triangleright_\omega b_i$ $(i = 1, \ldots, k)$ where the relation $\triangleright_\omega$ is given as follows. For all $M \in \Lambda_\omega(\mathfrak{M})^0$, $\sigma \in \mathbb{T}$, $a \in \mathfrak{M}_\sigma$ we have $M \triangleright_\omega a$ if and only if for all $(b_1, \ldots, b_k) \in (\mathfrak{M})_\sigma$

$$M\mathbf{C}(b_1) \ldots \mathbf{C}(b_k) =_{\beta\mu_\omega} \mathbf{C}(ab_1 \ldots b_k).$$

The ω-diagram of $\mathfrak{M}$ is now defined by

$$\mathcal{D}_\omega(\mathfrak{M}) = (\Lambda_\omega(\mathfrak{M}), \beta\mu_\omega).$$

So $\mathcal{D}_\omega(\mathfrak{M})$ is obtained from $\mathcal{D}(\mathfrak{M})$ by replacing the *constants* $\mathbf{C}(0), \mathbf{C}(1), \mathbf{C}(2), \ldots$ by *numerals* $\mathbf{C}(0), \mathbf{C}(1), \mathbf{C}(2), \ldots$.

Again by the same methods as used in sections 5.3 and 5.4 one can prove the following theorems.

5.6.5. THEOREM (Church-Rosser Theorem for ω-diagrams).
$\mathcal{D}_\omega(\mathfrak{M}) \models \mathbf{CR}$ *if and only if* $\mathfrak{M}$ *is quasi-extensional.* □

5.6.6. THEOREM (Compatibility Theorem for ω-diagrams).
If $\mathfrak{M}$ is quasi-extensional, then

$$M \vartriangleright_\omega a \;\&\; N \vartriangleright_\omega b \Rightarrow MN \vartriangleright_\omega ab. \square$$

In [Barendsen and Bezem 1994] ω-diagrams are used to transform models for simply typed (or first order) λ-calculi into models for polymorphic (or second order) λ-calculi. In this construction the Church-Rosser Theorem for ω-diagrams and the Compatibility Theorem for ω-diagrams play important roles.

The definitions of diagram and μ-reduction are inspired by [Kleene 1962]. In this paper Kleene introduced a diagram-like type-free λ-calculus that he uses to introduce a notion of λ-definability for partial functionals of higher types over ω. Subsequently he proves this notion of λ-definability to be equivalent to S1-S9 recursiveness. In the following we reformulate Kleene's results in the setting of diagrams.

Let $\mathfrak{M}$ be a quasi-extensional type structure with $\mathfrak{M}_0 = \omega$. The Kleene diagram $\mathcal{K}(\mathfrak{M})$ of $\mathfrak{M}$ is a type-free λ-calculus extending the $\lambda\mathbf{I}$-calculus. The Kleene diagram $\mathcal{K}(\mathfrak{M})$ of $\mathfrak{M}$ can be viewed as the $\lambda\mathbf{I}$-variant of the ω-diagram $\mathcal{D}_\omega(\mathfrak{M})$.

5.6.7. DEFINITION. The set $\Lambda^I_\omega(\mathfrak{M})$ is defined inductively as follows.

(i) $Var \subseteq \Lambda^I_\omega(\mathfrak{M})$
(ii) $a \in \mathfrak{M}_\sigma \;\&\; \sigma \neq 0 \Rightarrow \mathbf{C}(a) \in \Lambda^I_\omega(\mathfrak{M})$
(iii) $M, N \in \Lambda^I_\omega(\mathfrak{M}) \Rightarrow (MN) \in \Lambda^I_\omega(\mathfrak{M})$
(iv) $a \in Var, M \in \Lambda^I_\omega(\mathfrak{M}), x \in FV(M) \Rightarrow (\lambda x.M) \in \Lambda^I_\omega(\mathfrak{M})$.

The set $\Lambda^I_\omega(\mathfrak{M})$ is an extension of the set Λ^I of pure $\lambda\mathbf{I}$-terms. In Λ^I we have numerals $\mathbf{C}(0), \mathbf{C}(1), \mathbf{C}(2), \ldots$ defined by $\mathbf{C}(m) \equiv \lambda fx.f^{m+1}(x)$. So $\mathbf{C}(0) \equiv \lambda fx.fx$, $\mathbf{C}(1) \equiv \lambda fx.f(fx)$ etcetera.

An $M \in \Lambda^I_\omega(\mathfrak{M})$ is called normal if M is a β-normal form and does not contain any subterms of the form $\mathbf{C}(a)M_1 \ldots M_k$ with $a \in \mathfrak{M}_\sigma$, $\sigma = \sigma_1 \to \cdots \to \sigma_k \to 0 \neq 0$, $M_1, \ldots, M_k \in \Lambda^I_\omega(\mathfrak{M})^0$.

Analogous to the construction of the reductor μ it is possible to construct a reductor κ on $\Lambda^I_\omega(\mathfrak{M})$ consisting of all reduction rules

$$\mathbf{C}(a)M_1 \ldots M_k \to \mathbf{C}(ab_1 \ldots b_k)$$

with $a \in \mathfrak{M}_\sigma$, $\sigma = \sigma_1 \to \cdots \to \sigma_k \to 0 \neq 0$, $M_1, \ldots, M_k \in \Lambda^I_\omega(\mathfrak{M})^0$, $M_1, \ldots, M_k$ normal and $M_i \vartriangleright_\kappa b_i$ $(i = 1, \ldots, k)$, where of course the relation $\vartriangleright_\kappa$ is given as follows. For all $M \in \Lambda^I_\omega(\mathfrak{M})^0$, $\sigma \in \mathbb{T}$, $a \in \mathfrak{M}_\sigma$ we have $M \rhd_\kappa a$ if and only if for all $(b_1, \ldots, b_k) \in (\mathfrak{M})_\sigma$

$$M\mathbf{C}(b_1) \ldots \mathbf{C}(b_k) =_{\beta\kappa} \mathbf{C}(ab_1 \ldots b_k).$$

The Kleene diagram of $\mathfrak{M}$ is

$$\mathcal{K}(\mathfrak{M}) = (\Lambda^I_\omega(\mathfrak{M}), \beta\kappa).$$

Recall that we have taken $\mathfrak{M}$ to be quasi-extensional. Using a transfinite induction, similar to the one in the proof of $\mathbf{CR}(\beta\mu)$, one can show that $\mathbf{CR}(\beta\kappa)$. This proof is in fact simpler than the proof of $\mathbf{CR}(\beta\mu)$, because every stage in the transfinite construction of κ is a normal reductor, and therefore the First Church-Rosser Theorem applies.

The Kleene diagram $\mathcal{K}(\mathfrak{M})$ yields a notion of computability, called $\lambda\mathbf{I}$-computability, for partial functionals on $\mathfrak{M}$.

5.6.8. Definition. Let $\sigma \in \mathbb{T}$ and $\varphi : (\mathfrak{M})_\sigma \to_p \mathfrak{M}_0 = \omega$.

(i) Let $F \in (\Lambda^I)^0$. Then F $\lambda\mathbf{I}$-computes φ if for all $\vec{a} \in (\mathfrak{M})_\sigma$ and $m \in \omega$

$$\varphi(\vec{a}) \simeq m \iff F\mathbf{C}(\vec{a}) =_{\beta\kappa} \mathbf{C}(m).$$

(ii) φ is $\lambda\mathbf{I}$-computable if there exists $F \in (\Lambda^I)^0$ such that F $\lambda\mathbf{I}$-computes φ. $\square$

In particular we now have a notion of $\lambda\mathbf{I}$-computability for partial functionals on the full type structure $\mathfrak{M}(\omega)$ over ω. The main result of [Kleene 1962] can now be stated as follows.

5.6.9. Theorem (Kleene). *For every partial functional φ on $\mathfrak{M}(\omega)$: φ is $\lambda\mathbf{I}$-computable if and only if φ is S1-S9 recursive.* $\square$

Let $\mathfrak{M}$ be quasi-extensional with $\mathfrak{M}_0 = \omega$. The schemas S1-S9 (originally applied to the pure type structure over ω in [Kleene 1959]) can be applied to the type structure $\mathfrak{M}$, yielding a recursion theory of partial functionals on $\mathfrak{M}$, called the S1-S9 recursive functionals on $\mathfrak{M}$. Without any problems Kleene's proof of theorem 5.6.9 can be generalized to obtain the following generalization.

5.6.10. Theorem. *For every partial functional ϕ on $\mathfrak{M}$: ϕ is $\lambda\mathbf{I}$-computable if and only if ϕ is S1-S9 recursive.* $\square$

The ω-diagram $\mathcal{D}_\omega(\mathfrak{M})$ of $\mathfrak{M}$ also yields a notion of computability for partial functionals on $\mathfrak{M}$.

5.6.11. DEFINITION. Let $\sigma \in \mathbb{T}$ and $\varphi : (\mathfrak{M})_\sigma \to_p \omega$.

(i) Let $F \in \Lambda^0$. Then F λ-computes φ if for all $\vec{a} \in (\mathfrak{M})_\sigma$ and $m \in \omega$

$$\varphi(\vec{a}) \simeq m \iff F\mathbf{C}(\vec{a}) =_{\beta\mu} \mathbf{C}(m).$$

(ii) φ is λ-computable if there exists $F \in \Lambda^0$ such that F λ-computes φ. □

It is an open problem whether the notions of λ-computability and S1-S9 recursiveness for partial functionals on $\mathfrak{M}$ coincide. Kleene's proof of theorem 5.6.9 in [Kleene 1962] cannot be imitated without problems. Kleene uses a property of $\mathcal{K}(\mathfrak{M})$ that is not shared by $\mathcal{D}_\omega(\mathfrak{M})$. For $\mathcal{K}(\mathfrak{M})$ we have that every term having a normal form is strongly normalizing. This is not true for $\mathcal{D}_\omega(\mathfrak{M})$ as it is a property typical of the $\lambda\mathbf{I}$-calculus. Using this property Kleene assigns, to every $M \in \Lambda^I_\omega(\mathfrak{M})^0$ a so-called reduction tree $\mathsf{T}(M)$ with the property that if M has a normal form, then $\mathsf{T}(M)$ is well-founded. Then $\mathsf{T}(M)$ corresponds with an ordinal number that can be seen as the length of the 'computation' starting with M. We have not been able to construct a tree-assignment for $\mathcal{D}_\omega(\mathfrak{M})$ with a similar property. However, we still feel that something like this is possible, and therefore we conjecture: for partial functionals on $\mathfrak{M}$ the notions of λ-computability and S1-S9 recursiveness coincide.

Bibliography

[Barendregt 1984] H.P. Barendregt
The Lambda-Calculus-Its Syntax and Semantics
Studies in Logic 103
North Holland Amsterdam 1984

[Barendsen and Bezem 1994] E. Barendsen and M. Bezem
Polymorphic Extensions of Simple Type Structures
Technical Report CSI-R9412
Computing Science Institute
Nijmegen University

[Barwise 1977] J. Barwise (ed.)
Handbook of Mathematical Logic
Studies in Logic 90
North Holland Amsterdam 1977

[Bergstra 1976] J. Bergstra
Computability and continuity in finite types
Ph.D.Thesis Utrecht 1976

[Bethke 1988] I. Bethke
Notes on partial combinatory algebras
Ph.D.Thesis Amsterdam 1988

[Bezem 1986] M.Bezem
Bar recursion and functionals of finite type
Ph.D.Thesis Utrecht 1986

[Fenstad 1980] J.E. Fenstad
General recursion
Perspectives in Mathematical Logic
Springer Verlag 1980

[Gandy and Hyland 1977] R.O. Gandy and J.M.E. Hyland
Computable and recursively countable functions of higher type

Logic Colloquium 76
North Holland Amsterdam 1977

[Goldblatt 1984] R. Goldblatt
Topoi-the categorical analysis of logic
Studies in Logic 98
North Holland Amsterdam 1984

[Hinman 1978] P.G. Hinman
Recursion-theoretic Hierarchies
Perspectives in Mathematical Logic
Springer Verlag 1978

[Kechris and Moschovakis 1977] A.S. Kechris and Y.N. Moschovakis
Recursion in higher types
In [Barwise 1977]

[Kleene 1959] Recursive functionals and quantifiers of finite type I
Trans. Amer. Math. Soc. 91

[Kleene 1962] S.C. Kleene
Lambda-definable functionals of finite type
Fund. Math. 50

[Kleene 1963] Recursive functionals and quantifiers of finite type II
Trans. Amer. Math. Soc. 108

[Klop 1980] J.W. Klop
Combinatory reduction systems
Mathematical Center Tracts 129 Amsterdam 1980

[Kreisel 1959] G. Kreisel
Interpretation of analysis by means of constructive functionals of finite types
Constructivity in mathematics (Heyting, ed.)
North Holland Amsterdam

[Kuper 1994] J. Kuper
Partiality in logic and computation-Aspects of undefinedness
Ph.D.Thesis Twente 1994

[MacLane 1972] S. MacLane
Categories for the working mathematician
Springer Verlag 1972

[Moldestad 1977] J. Moldestad
Computations in higher types

Lecture Notes in Mathematics 574
Springer Verlag 1977

[Moschovakis 1969] Y.N. Moschovakis
Axioms for computation theories-first draft
Logic Colloquium 69 (Gandy and Yates eds.)
North Holland Amsterdam 1971

[Odifreddi 1989] P. Odifreddi
Classical Recursion Theory
Studies in Logic 125
North Holland Amsterdam 1989

[Lambek and Scott 1986] J. Lambek and P.J. Scott
Introduction to higher order categorical logic
Cambridge Studies in mathematics 7
Cambridge University Press 1986

[Platek 1966] R.A. Platek
Foundations of recursion theory
Ph.D.Thesis Stanford 1966

[Plotkin 1977] G.D. Plotkin
LCF considered as a programming language
Theor. Comp. Sci. 5

[Spector 1962] C. Spector
Provably recursive functionals of analysis: a consistency proof of analysis by an extension of principles formulated in current intuitionistic mathematics
Recursive function theory (Dekker), Proc. of symposia in pure math. V
Am. Math. Soc. Providence, RI

[Troelstra 1973] A.S. Troelstra
Metamathematical Investigations of intuitionistic arithmetic and analysis
Lecture Notes in Mathematics 344
Springer Verlag 1973

[Zucker 1971] Z. Zucker
Proof theoretic studies of systems of iterative inductive definitions and subsystems of analysis
Ph.D.Thesis Stanford 1971

Samenvatting

Dit proefschrift gaat over simpel-getypeerde λ-calculi en over modellen voor simpel-getypeerde λ-calculi. Verder zijn we geïnteresseerd in recursie-theorieën op type-structuren. Deze worden gebruikt om modellen voor bepaalde simpel-getypeerde λ-calculi te construeren.

Hoofdstuk 1 gaat over de simpel-getypeerde λ-calculi zelf. We zijn uitsluitend geïnteresseerd in simpel-getypeerde λ-calculi die ontstaan uit de pure simpel-getypeerde λ-calculus $\lambda^{\to}$ door toevoeging van constanten en δ-reductie. Door middel van δ-reductie kunnen externe functies op λ-termen intern gemaakt worden. Is f zo'n externe functie op λ-termen, dan wordt een constante $\mathbf{C}$ gekozen als representant voor f, en met behulp van δ-reductie wordt het gedrag van f door $\mathbf{C}$ nagebootst:

$$\mathbf{C}M \to f(M).$$

Door middel van 'currying' kunnen ook functies met meerdere argumenten intern gemaakt worden:

$$\mathbf{C}M_1 \dots M_m \to f(M_1, \dots, M_m).$$

Door toevoeging van constanten en δ-reductie aan $\lambda^{\to}$ kunnen λ-calculi met bijvoorbeeld primitieve recursoren ($\lambda\mathsf{T}$), barrière-recursoren ($\lambda\mathsf{B}$) of dekpunt-combinatoren ($\lambda\mathsf{Y}$) worden verkregen.

Een ander gebruik van constanten is de representatie van natuurlijke getallen. Door toevoeging van een rij $\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2, \dots$ van type-0 constanten wordt een numeral systeem verkregen. Dit sluit beter aan bij de intuïtie dat natuurlijke getallen objecten van type 0 zijn, dan bijvoorbeeld het gebruik van getypeerde Church-numerals $\lambda fx.f^n(x)$ in $\lambda^{\to}$.

Door toevoeging aan $\lambda^{\to}$ van dekpunt-combinatoren $\mathbf{Y}_\sigma$, natuurlijke getallen $\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2, \dots$, successor $\mathbf{S}^+$ en een gelijkheidstest $\mathbf{\Delta}$ op de $\mathbf{c}_n$-en wordt de simpel-getypeerde λ-calculus $\lambda\mathsf{Y}(\mathcal{N})$ verkregen. Het systeem $\lambda\mathsf{Y}(\mathcal{N})$ is op enkele kleine details na identiek aan Plotkin's PCF ([Plotkin 1977]). In het systeem $\lambda\mathsf{Y}(\mathcal{N})$ kan elke recursieve functie berekend worden.

In hoofdstuk 2 bestuderen we modellen voor simpel-getypeerde λ-calculi. Een model voor een simpel-getypeerde λ-calculus Σ is een structuur, bestaande uit een type-structuur $\mathfrak{M}$ tesamen met een interpretatie van de Σ-termen in $\mathfrak{M}$, met de eigenschap dat $\mathfrak{M}$ op een bepaalde manier is afgesloten onder afleidbaarheid in Σ. Een verzwakking van het model-begrip is het begrip 'algebra'. Een Σ-algebra is een structuur, bestaande uit een type-structuur $\mathfrak{M}$ en een interpretatie van de Σ-termen in $\mathfrak{M}$, zodat elke in Σ afleidbare gelijkheid geldig is in die structuur.

Een variant van het begrip 'model' is het begrip 'quasi-model'. In gewone modellen wordt λ-calculus-gelijkheid geïnterpreteerd door gewone gelijkheid in type-structuren, maar in quasi-modellen wordt gelijkheid in de λ-calculus geïnterpreteerd door extensionele gelijkheid in type-structuren. Het bijbehorende verzwakte begrip 'quasi-algebra' blijkt tevens een verzwakking te zijn van het gewone algebra-begrip, terwijl het begrip 'quasi-model' geen verzwakking blijkt te zijn van het gewone model-begrip.

In hoofdstuk 3 wordt een algemeen procédé beschreven om, uitgaande van cartesisch gesloten categorieën, algebra's en modellen voor simpel-getypeerde λ-calculi te construeren. Uitgaande van de categorie **Mon** van cpo's (complete partiële ordeningsstructuren) met monotone afbeeldingen, levert dit procédé een $\lambda\mathsf{Y}(\mathcal{N})$-model op, met als onderliggende type-structuur de type-structuur **MON**, die uit de platte cpo $\omega_\perp$ ontstaat door herhaaldelijke toepassing van de monotone-functie-ruimte-operatie.

Hoofdstuk 4 gaat over simpel-getypeerde λ-calculi met enumeratoren. Een enumerator is een gesloten λ-term die alle gesloten λ-termen van een bepaald type modulo conversie enumereert. Met behulp van δ-reductie is het eenvoudig systemen met enumeratoren te construeren. Het standaard-voorbeeld van zo'n systeem met enumeratoren is $\lambda\mathsf{E}(\mathcal{N})$. Dit systeem bevat voor elk type σ een enumerator $\mathbf{E}_\sigma$ die alle gesloten λ-termen van type σ modulo $\lambda\mathsf{E}(\mathcal{N})$-conversie enumereert. Verder bevat $\lambda\mathsf{E}(\mathcal{N})$ natuurlijke getallen $\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2, \ldots$ van type 0, een successor $\mathbf{S}^+$ en een gelijkheidstest $\mathbf{\Delta}$ op de $\mathbf{c}_n$-en. We bewijzen dat $\lambda\mathsf{E}(\mathcal{N})$ dekpunt-combinatoren bevat. Daarom bestaat er een vertaling van $\lambda\mathsf{Y}(\mathcal{N})$ in $\lambda\mathsf{E}(\mathcal{N})$, en is elke recursieve functie $\lambda\mathsf{E}(\mathcal{N})$-berekenbaar. We bewijzen ook dat het systeem $\lambda\mathsf{Y}(\mathcal{N})$ geen enumeratoren bevat, zodat $\lambda\mathsf{E}(\mathcal{N})$ in zekere zin sterker is dan $\lambda\mathsf{Y}(\mathcal{N})$, hoewel beide systemen precies de recursieve functies berekenen.

Het hoofdresultaat van dit proefschrift is de constructie van een model voor $\lambda\mathsf{E}(\mathcal{N})$. Omdat er een vertaling bestaat van $\lambda\mathsf{Y}(\mathcal{N})$ in $\lambda\mathsf{E}(\mathcal{N})$, moet een $\lambda\mathsf{E}(\mathcal{N})$-model gezocht worden tussen de $\lambda\mathsf{Y}(\mathcal{N})$-modellen. Zoals we zojuist gezegd hebben is de type-structuur **MON** een $\lambda\mathsf{Y}(\mathcal{N})$-model. We bewijzen dat **MON** ook een $\lambda\mathsf{E}(\mathcal{N})$-model is. Hiertoe wordt eerst een recursie-theorie op **MON** geconstrueerd. Dit is een collectie van totale, monotone functionalen op **MON** met bepaalde eigenschappen, waaronder een Enumeratie-eigenschap. Voor deze recursie-theorie gelden een Eerste Recursie-stelling en een Tweede Recursie-stelling. Uit de Tweede Recursie-stelling volgt dan dat **MON** een $\lambda\mathsf{E}(\mathcal{N})$-model is.

In feite bewijzen we een algemener resultaat. We voeren een algemeen begrip 'recursie-theorie op een type-structuur' in. Laat $\mathfrak{M}$ een quasi-extensionele type-structuur zijn, en laat $\mathcal{R}$ een recursie-theorie op $\mathfrak{M}$ zijn. Dan gelden voor $\mathcal{R}$ een Eerste Recursie-stelling en een Tweede Recursie-stelling. Uit de Eerste Recursie-stelling volgt dat $\mathfrak{M}$ een quasi-$\lambda\mathsf{Y}(\mathcal{N})$-model is, en uit de Tweede Recursie-stelling volgt dat $\mathfrak{M}$ een quasi-$\lambda\mathsf{E}(\mathcal{N})$-model is.

Andersom is het zo dat als $\mathfrak{M}$ een quasi-extensioneel quasi-$\lambda\mathsf{E}(\mathcal{N})$-model is, dan bestaat er een recursie-theorie $\mathcal{R}$ op $\mathfrak{M}$ zodat de $\lambda\mathsf{E}(\mathcal{N})$-definiëerbare objecten in $\mathfrak{M}$ precies overeenkomen met de $\mathcal{R}$-recursieve objecten in $\mathfrak{M}$.

In hoofdstuk 5 associëren we met elke type-structuur $\mathfrak{M}$ een simpel-getypeerde λ-calculus $\lambda(\mathfrak{M})$. Het systeem $\lambda(\mathfrak{M})$ heet het diagram van $\mathfrak{M}$. Dit begrip 'diagram' is gebaseerd op een reductie-relatie die gebruikt werd in [Kleene 1962] om een begrip van λ-berekenbaarheid voor hogere-type functionalen in te voeren.

Het verband tussen hoofdstuk 5 en de eerdere hoofdstukken (in het bijzonder hoofdstuk 2) is het begrip 'quasi-extensionaliteit'. Het hoofdresultaat van hoofdstuk 5 is de karakterizering van de klasse van alle quasi-extensionele type-structuren als precies die type-structuren $\mathfrak{M}$ waarvoor geldt $\lambda(\mathfrak{M}) \models \mathbf{CR}$.

Curriculum Vitae

- Geboren 26-1-1966 te Uden.
- VWO, Rivendel College Uden, 1978-1984.
- Wiskunde-studie aan de Katholieke Universiteit Nijmegen, 1984-1989, specialistische variant met als specialisaties grondslagen van de wiskunde en grondslagen van de informatica, afgestudeerd 5 oktober 1989 (cum laude).
- Onderzoeker in opleiding aan de Katholieke Universiteit Nijmegen, afdeling Grondslagen van de vakgroep Informatica, 1989-1993.