MATHEMATISCHES INSTITUT
DER LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Diplomarbeit

On the Computational Content of Choice Axioms

vorgelegt von
Simon Huber

MATHEMATISCHES INSTITUT
DER LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Diplomarbeit

On the Computational Content of Choice Axioms

vorgelegt von
Simon Huber

Betreuer Prof. Dr. Helmut Schwichtenberg
Abgabetermin 14. Januar 2010

Abstract

The present thesis studies the computational content of the axiom of countable choice and the axiom of dependent choice, as well as some of their classical equivalents. This content can be obtained by various different recursion schemes — here we turn our attention to the following definition principles: bar recursion of Spector, a variant thereof due to Kohlenbach, open recursion of Berger, the functional introduced by Berardi, Bezem, and Coquand, and the related modified bar recursion of Berger and Oliva. We consider the following aspects. First, we introduce the partial continuous functionals using the information systems of Scott, and, building on that, the Kleene-Kreisel continuous functionals. Moreover, we develop a term system with a corresponding semantic. Second, we show that the Kleene-Kreisel continuous functionals are a model of the recursion schemes above. This is proved using proof principles closely related to the recursion schemes. Third, we turn to the question of interdefinability of the schemes. Finally, we show how to extract programs, which use one of the recursion schemes, from proofs using classical choice principles. Here we restrict ourselves to approaches using a combination of modified realizability and A-translation. This includes modified bar recursion, open recursion, and variants thereof.

Zusammenfassung

Die vorliegende Arbeit befasst sich mit dem rechnerischen Gehalt des abzählbaren Auswahlaxioms und des Axioms der abhängigen Auswahl, sowie dazu klassisch äquivalenter Prinzipien. Dieser Gehalt kann durch verschiedene Rekursionsschemata ermittelt werden — wir beschränken uns hier auf folgende Definitionsprinzipien: Bar-Rekursion nach Spector, deren Variante nach Kohlenbach, offene Rekursion nach Berger, das von Berardi, Bezem und Coquand eingeführte Funktional, sowie die verwandte, von Berger und Oliva eingeführte, modifizierte Bar-Rekursion. Wir untersuchen hier die folgenden Aspekte. Erstens führen wir die partiell stetigen Funktionale durch Informationssysteme von Scott ein und, darauf basierend, die Kleene-Kreisel stetigen Funktionale. Weiter entwickeln wir ein Termsystem und eine dazugehörige Semantik. Zweitens zeigen wir, dass die Kleene-Kreisel stetigen Funktionale ein Modell für unsere Rekursionsschemata sind. Dies wird durch Beweisprinzipien gezeigt, welche eng mit den Rekursionsschemata verwandt sind. Drittens wenden wir uns der Frage der Interdefinierbarkeit der verschiedenen Rekursionsschemata zu. Schließlich zeigen wir, wie man aus Beweisen, die klassische Auswahlprinzipien verwenden, Programme extrahieren kann, welche eines der obigen Schemata verwendet. Hierbei beschränken wir uns auf Ansätze, die mit einer Kombination aus A-Übersetzung und Realisierbarkeitsinterpretation behandelbar sind. Dies beinhaltet die modifizierte Bar-Rekursion sowie die offene Rekursion und deren Varianten.

Danksagung

An erster Stelle danke ich meinem Betreuer Prof. Dr. Helmut Schwichtenberg für die stete Unterstützung und das ideale Arbeitsumfeld. Nicht nur war – und ist – er ein hervorragender Lehrer für mich, sondern er hatte auch immer ein offenes Ohr für meine Fragen und Anmerkungen.

Desweiteren möchte ich mich bei meinen jetzigen und früheren Bürokollegen für die vielen erhellenden Gespräche und unterhaltsamen Momente bedanken. Ebenso danke ich den übrigen Mitgliedern der Logik-Arbeitsgruppe für die schöne Atmosphäre. Bei Max Huber, Basil Karádais und Florian Ranzi bedanke ich mich für das Korrekturlesen meiner Arbeit.

Für Diskussionen zum Thema möchte ich mich auch bei Prof. Dr. Ulrich Berger, Dr. Philipp Gerhardy, Dr. Paulo Oliva und Priv.-Doz. Dr. Peter Schuster bedanken. Herrn Schuster danke ich dafür, dass er mich auf das Lemma bzw. Induktionsprinzip von Tukey-Teichmüller aufmerksam gemacht hat, was zur Formulierung der erweiterten Updateinduktion führte.

Schließlich möchte ich mich bei meinen Freunden und meiner Familie bedanken dafür, dass sie immer für mich da waren. Meinen Eltern danke ich auch für die andauernde Unterstützung, die mir das Studium erst ermöglicht hat.

Contents

Introduction

A fundamental question of proof theory being of interest to mathematicians and computer scientists is to determine and classify the computational content of a formal proof.

Suppose we have a formal proof of a statement of the form $\forall n \exists m A(n,m)$ with A a quantifier-free formula (this may be regarded as a specification of a program). Then we are interested in a method for extracting an algorithm or program p from our given proof such that $\forall n A(n, p(n))$ holds. Moreover, we are interested in the complexity of p.

If the formal proof is constructive in nature, this is directly possible via the so-called Curry-Howard correspondence, i.e., the proofs-as-programs paradigm. The idea of "unwinding" the constructive content of prima facie non-constructive proofs goes back to Kreisel. This was formulated as an alternative to Hilbert's consistency program:

> "There is a general program which does not seem to suffer the defects of the consistency program: To determine the constructive (recursive) content or the constructive equivalent of the non-constructive concepts and theorems used in mathematics, particularly arithmetic and analysis." [**25**, p. 155]

One aim of the thesis is to explore the computational content of choice principles in a classical context, more precisely, the axiom of countable and dependent choice, as well as classical equivalent principles.

The axiom of (countable) choice plays a crucial role in classical analysis. Together with the principle of excluded middle, it implies the full comprehension axiom. This allows to introduce sets of natural numbers impredicatively: Given a formula $A(n)$, comprehension enables us to form the set $\{n \in \mathbb{N} \mid A(n)\}$ — even if the formula A contains quantification over sets of natural numbers, i.e., already refers to the entity of *all* sets of naturals.

If one interprets the axiom of choice constructively via the Brouwer-Heyting-Kolmogorov interpretation of the logical constants, then it is valid. However, the negative translation of the axiom of choice is not provable intuitionistically and so the axiom of choice is apparently weaker in an intuitionistic setting. Thus, from a constructive point of view, the (classical) axiom of choice turns out to be problematic.

All this suggests that a computational interpretation of classical choice principles is not straightforward. This was first achieved in 1962 by Spector [**42**], who extended Gödel's Dialectica (or functional) interpretation of arithmetic [**18**] to classical analysis by an extension of Gödel's T with a new principle of recursion, the so-called *bar recursion*. There have been various other methods, most notably Girard's interpretation of second order arithmetic using the polymorphic λ-calculus [**17**], but we now concentrate on approaches relevant to our investigations.

In [**2**], Berardi, Bezem, and Coquand proposed a different interpretation of classical countable choice and dependent choice which resulted in a more demand-driven algorithmic content than Spector's interpretation. Their method builds on a variation of realizability opposed to the Dialectica interpretation used by Spector. Building on that, Berger [**4**] introduced open recursion which realizes open induction — a fragment of an induction principle by Raoult [**34**], which is classically equivalent to dependent choice. Moreover, he proposed a special instance of open recursion, namely update recursion, and the corresponding update induction. It turned out that the functional of Berardi, Bezem, and Coquand is a special case of update recursion.

In this thesis we propose yet another definition and proof principle called extended update recursion and induction respectively, which lies between open and update recursion and induction. This allows to obtain the computational content of the axiom of choice along the lines of the update recursion/induction approach.

Later, Berger and Oliva [**6, 7**] introduced a variant of Spector's bar recursion, which they dubbed modified bar recursion and allowed for an easier correctness and termination proof. Here we present how one can obtain the computational content of classical choice principles using the approaches based on modified bar recursion and variants of open recursion.

All of the aforementioned recursion schemes come with different variants and flavors. Two natural questions arise: (a) Are there any mathematical models of these definition principles, i.e., are they consistent? (b) How do these recursion schemes relate w.r.t. interdefinability?

This thesis tackles both questions. Concerning (a), we introduce the Kleene-Kreisel continuous functionals and show that they are a model of the recursion schemes. For this, we develop the theory of partial continuous functionals based on coherent Scott information systems up to the density theorem; the Kleene-Kreisel functionals are then equivalence classes of certain partial continuous functionals. Moreover, we formulate a flexible term language together with an operational and adequate denotational semantics. Our treatment is kept very general and thus is also of general interest.

Concerning (b), many questions have been answered in [**7**]. We present some of these results and extend on some relations.

Outline of the Contents

This diploma thesis is organized as follows. Chapter 1 introduces a general typed term language and the formal systems we use. Moreover, we formulate a reduction relation on terms and show its confluence. In Chapter 2 we develop the general theory of coherent information systems and, building on this, we introduce the partial continuous functionals. Then we define the denotation of a term as a partial continuous functional and an operational semantics which is computationally adequate w.r.t. the denotational semantics. Next, we single out the total among the partial functionals, prove Kreisel's Density Theorem, and introduce the model of the Kleene-Kreisel continuous functionals which forms a model of classical arithmetic plus dependent choice. Chapter 3 defines all recursion schemes and the corresponding axiom schemes, shows the validity of the recursion principles in the Kleene-Kreisel continuous functionals, and studies the interdefinability of these recursion principles. Moreover, we give an algorithm for the fan functional using two of our schemes. In the last chapter we show how one can extract the computational content using the discussed schemes from a proof using choice axioms; here we restrict ourselves to methods using variants of open recursion and modified bar recursion.

Preliminaries

The set of natural numbers $\{0, 1, 2, \dots\}$ is denoted by $\mathbb{N}$. We write $X \subseteq^{fin} Y$ if X is a finite subset of Y.

If R is a binary relation on a set X, we usually write R in infix notation. The transitive closure of R is denoted by R^+ and the reflexive and transitive closure by R^*. We define $R^{\leq m} := \bigcup_{0 \leq n \leq m} R^n$, where $R^0 := \{(x, x) \mid x \in X\}$ and $R^{n+1} := \{(x, y) \mid \exists z \in X(xR^n z \wedge zRy)\}$.

Whenever we introduce a metavariable for some sort of objects, we tacitly assume that all primed and indexed variants range over the same sort of objects.

We make extensive use of the arrow notation for finite lists as follows. An arrow over a symbol stands for a finite list of objects of the same sort as the symbol, e.g., $\vec{x}$ stands for a finite list $x_1, \dots, x_n$. The length of a finite sequence $\vec{x} = x_1, \dots, x_n$ is n and denoted by $|\vec{x}|$. Note that this includes the empty sequence, denoted by $\langle\rangle$, if $n = 0$. As usual, expressions involving the arrow notation should be read componentwise. For instance, if not stated otherwise, $\vec{x}R\vec{y}$ should be read as $|\vec{x}| = |\vec{y}|$ and $x_1Ry_1, \dots, x_nRy_n$ for a binary relation R; if X is a set, $\vec{x} \in X$ means $x_1 \in X, \dots, x_{|\vec{x}|} \in X$. However, if clear from the context or stated explicitly, there will be exceptions to this rule. Sometimes we identify $\vec{x}$ with the set of its components $\{x_1, \dots, x_{|\vec{x}|}\}$. E.g., we write $x \in \vec{x}$ if x is a component in the list $\vec{x}$, or $\mathsf{FV}(\mathcal{E}) \subseteq \vec{x}$ if the free variables of $\mathcal{E}$ are among $x_1, \dots, x_{|\vec{x}|}$.

We identify α-equivalent formulas, terms, and types. The set of free variables of an expression E (i.e., E a formula of term) is defined as usual and denoted by $\mathsf{FV}(E)$; if $\mathcal{E}$ is a set or list of expressions we set $\mathsf{FV}(\mathcal{E}) = \bigcup_{E \in \mathcal{E}} \mathsf{FV}(E)$. An expression is called *closed* if it has no free variables.

A *substitution*, denoted by ξ, is a finite mapping from variables to terms preserving the type, i.e., it is a finite set of pairs (x_i, M_i) $(i = 1, \dots, n)$, where the x_i's are distinct and x_i has the same type as M_i. We write $E[\vec{M}/\vec{x}]$ for this substitution applied to an expression E. The application of such a substitution to a term or formula is defined "capture-free", that is, it operates on terms and formulas by simultaneously replacing x_i by M_i, renaming bound variables whenever necessary to avoid that free variables in the M_i become bound. If an expression E has been introduced as $E(\vec{x})$ ($\vec{x}$ distinct variables), then $E(\vec{M})$ denotes $E[\vec{M}/\vec{x}]$. The same conventions apply to type substitutions. We assume that the reader is acquainted with basic first-order unification, in particular, most general unifiers as presented in, e.g., [**46**, p. 33 f.].

We use a mild form of the dot-notation: A dot stands for a pair of parentheses opening at the dot and extending to the right as far as syntactically possible. E.g., $\forall x.A \to B$ stands for $\forall x(A \to B)$.

Concerning iterated implications, we write $A_1 \to A_2 \to \ldots \to A_n \to B$ or just $\vec{A} \to B$ with $\vec{A} = A_1, \ldots, A_n$ for $A_1 \to (A_2 \to \ldots \to (A_n \to B) \ldots)$; for arrow types we write $\rho_1 \to \rho_2 \to \cdots \to \rho_n \to \sigma$ or just $\vec{\rho} \to \sigma$ with $\vec{\rho} = \rho_1, \ldots, \rho_n$ for $\rho_1 \to (\rho_2 \to \cdots \to (\rho_n \to \sigma) \ldots)$.

We write IH (SIH) as an abbreviation for (side) induction hypothesis. Sometimes we write "Ind($\mathcal{E}$)" for "Induction on $\mathcal{E}$", where $\mathcal{E}$ is usually inductively defined. If $\mathcal{E}$ is more than one expression, we tacitly mean an induction on the "maximum" (only if this makes sense). Moreover, "Cases($\mathcal{E}$)" abbreviates "case distinction on $\mathcal{E}$".

CHAPTER 1

Syntax

In this chapter we introduce all syntactical concepts used in the thesis. This includes a typed term language similar to Plotkin's *PCF* [**33**]. However, our approach is based on computation rules, which are somehow more flexible than fixed-point operators, and also includes *PCF* and Gödel's T [**18**] as special cases. Moreover, our language features a rich type system in the sense that it has so called "free algebras" as base types. Furthermore we give a reduction relation for our term system and prove its confluence. In the last part of the present chapter we introduce all formal systems, including Heyting and Peano Arithmetic in higher types, which are used in the rest of the thesis.

Apart from minor differences, the definitions of terms and types are taken from [**37**] and [**35**].

1. Types

Our type system provides general datatypes as base types called "algebras" or "μ-types". For example, the type for the natural numbers can be defined as $\mu\alpha(\alpha, \alpha \to \alpha)$. Here, the list $\alpha, \alpha \to \alpha$ stands for the generation principles of the natural numbers — the zero and successor operations. In addition, we also allow simultaneously defined algebras.

Let TyVar be a countably infinite set of type variables.

Definition 1.1 (Types, Constructor Types). *Types* (denoted by ρ, σ, $\tau \in \mathsf{Ty}$) and *constructor types* (denoted by $\kappa \in \mathsf{KT}(\vec{\alpha})$ for $\vec{\alpha} \in \mathsf{TyVar}$ distinct type variables) are inductively defined by the clauses:

$$\frac{\vec{\rho}, \vec{\sigma}_1, \dots, \vec{\sigma}_n \in \mathsf{Ty} \quad \vec{\alpha} \in \mathsf{TyVar} \text{ distinct}}{\vec{\rho} \to (\vec{\sigma}_1 \to \alpha_{j_1}) \to \dots \to (\vec{\sigma}_n \to \alpha_{j_n}) \to \alpha_j \in \mathsf{KT}(\vec{\alpha})}$$

$$\frac{\vec{\kappa} \in \mathsf{KT}(\vec{\alpha})}{(\mu\vec{\alpha}.\vec{\kappa})_j \in \mathsf{Ty}} \ (|\vec{\kappa}| \geq 1, 1 \leq j \leq |\vec{\alpha}|) \qquad \frac{\rho, \sigma \in \mathsf{Ty}}{\rho \to \sigma \in \mathsf{Ty}}$$

A type of the form $(\mu\vec{\alpha}.\vec{\kappa})_j$ is called an *algebra*, μ-, *ground*-, or *base type*. Types of the form $\rho \to \sigma$ are called *arrow types*. We say that the algebras $\vec{\mu} = \mu\vec{\alpha}.\vec{\kappa}$ are *simultaneously defined*. In $(\mu\vec{\alpha}.\vec{\kappa})_j$, $\vec{\alpha}$ are bound and we identify types up to α-equivalence. The set of all μ-types is denoted by $\mu\mathsf{Ty}$ and we assume that the letter μ ranges over $\mu\mathsf{Ty}$. The *parameter types* of $\mu = (\mu\vec{\alpha}.\vec{\kappa})_j$ are the members of all $\vec{\rho}$ appearing in the constructor types $\kappa \in \vec{\kappa}$ of the form

$$\kappa = \vec{\rho} \to (\vec{\sigma}_1 \to \alpha_{j_1}) \to \dots \to (\vec{\sigma}_n \to \alpha_{j_n}) \to \alpha_j.$$

A type is called *finitary*, if it is a base type $\mu = (\mu\vec{\alpha}.\vec{\kappa})_j$ and for all $1 \le i \le |\vec{\kappa}|$ with $\kappa_i = \vec{\rho} \to (\vec{\sigma}_1 \to \alpha_{j_1}) \to \cdots \to (\vec{\sigma}_n \to \alpha_{j_n}) \to \alpha_j$ all members of $\vec{\rho}$ are finitary and all $\vec{\sigma}_k$ are empty ($1 \le k \le n$).

Examples. We now give some examples of base types. Let $\rho, \sigma \in \mathsf{Ty}$.

$\mathbf{N} := \mu\alpha(\alpha, \alpha \to \alpha)$	natural numbers,
$\mathbf{B} := \mu\alpha(\alpha, \alpha)$	booleans,
$\rho^* := \mu\alpha(\alpha, \rho \to \alpha \to \alpha)$	lists of elements of type ρ,
$\rho \times \sigma := \mu\alpha(\rho \to \sigma \to \alpha)$	product type,
$\rho + \sigma := \mu\alpha(\rho \to \alpha, \sigma \to \alpha)$	sum type,
$\mathcal{O} := \mu\alpha(\alpha, \alpha \to \alpha, (\mathbf{N} \to \alpha) \to \alpha)$	countable ordinals.

The examples above are all non-simultaneously defined algebras. For completeness, we also give examples of simultaneously defined algebras:

$$(\mathbf{Ev}, \mathbf{Od}) := \mu\alpha, \beta.(\alpha, \beta \to \alpha, \alpha \to \beta) \quad \text{even and odd numbers,}$$

$$(\mathbf{T}(\rho), \mathbf{Ts}(\rho)) := \mu\alpha, \beta.(\rho \to \alpha, \beta \to \alpha, \beta, \alpha \to \beta \to \beta) \quad \text{trees and tree lists.}$$

Definition 1.2. Let $\mathcal{B} \subseteq \mathsf{Ty}$. We call $\mathcal{B}$ a *system of types* if $\mathcal{B} \cap \mu\mathsf{Ty} \neq \emptyset$, $\mathcal{B}$ is closed under arrow types (i.e., $\rho, \sigma \in \mathcal{B}$ implies $\rho \to \sigma \in \mathcal{B}$), and for all $(\mu\vec{\alpha}.\vec{\kappa})_j \in \mathcal{B}$ and $1 \le l \le |\vec{\alpha}|$, also $(\mu\vec{\alpha}.\vec{\kappa})_l \in \mathcal{B}$ and for all $\kappa \in \vec{\kappa}$, if $\kappa = \vec{\rho} \to (\vec{\sigma}_1 \to \alpha_{j_1}) \to \cdots \to (\vec{\sigma}_n \to \alpha_{j_n}) \to \alpha_j$, then $\vec{\rho}, \vec{\sigma}_1, \ldots, \vec{\sigma}_n \in \mathcal{B}$.

In the sequel we assume that a system of types $\mathcal{B}$ is given and that all types range over $\mathcal{B}$.

2. Terms

For each $\vec{\mu} = \mu\vec{\alpha}.\vec{\kappa}$ and for each $1 \le i \le |\vec{\kappa}|$ we have a *constructor* $C_i^{\vec{\mu}}$ of type $\kappa_i[\vec{\mu}/\vec{\alpha}]$. We assume that C ranges over constructors and we write C^σ to indicate that C is of type σ.

Examples. For the naturals $\mathbf{N}$, we have two constructors denoted by $0^{\mathbf{N}}$ (zero) and $S^{\mathbf{N} \to \mathbf{N}}$ (successor). The booleans $\mathbf{B}$ have two constructors $\mathrm{tt}^{\mathbf{B}}$ (true) and $\mathrm{ff}^{\mathbf{B}}$ (false). Further examples are:

$(\mathrm{inl}_{\rho,\sigma})^{\rho \to \rho+\sigma}, (\mathrm{inr}_{\rho,\sigma})^{\sigma \to \rho+\sigma}$	left and right injection,
$(\mathrm{pair}_{\rho,\sigma})^{\rho \to \sigma \to \rho \times \sigma}$	pairing,
$(\mathrm{nil}_\rho)^{\rho^*}, (\mathrm{cons}_\rho)^{\rho \to \rho^* \to \rho^*}$	list constructors.

An *arity* is a list of types and is denoted with an extra pair of outer parentheses, i.e., by $(\sigma_1, \ldots, \sigma_n)$. A *defined constant of type* σ is a symbol D together with a type σ and a fixed arity $\mathsf{ar}(D)$ such that $\mathsf{ar}(D) = (\sigma_1, \ldots, \sigma_n)$ where $\sigma = \vec{\sigma} \to \rho$ for some type ρ. It is convenient to assign an arity to constructors by $\mathsf{ar}(C^{\vec{\tau} \to \mu}) := (\vec{\tau})$. Let $\mathcal{D}$ be a set of defined constants. In the sequel we assume that D ranges over $\mathcal{D}$. Again, a superscript D^σ indicates that D is of type σ.

For each type σ we assume a countably infinite set Var_σ of variables of type σ denoted by $x^\sigma, y^\sigma, z^\sigma$. Let Var be the set of all variables.

DEFINITION 2.1 (($\mathcal{B}$, $\mathcal{D}$)-terms).

$$M, N ::= x^\sigma \mid (\lambda x^\rho M^\sigma)^{\rho\to\sigma} \mid (M^{\rho\to\sigma} N^\rho)^\sigma \mid C^\sigma \mid D^\sigma$$

Terms are identified up to α-equivalence. We write M^σ or $M\colon \sigma$ to indicate that M is a term of type σ. We usually assume that all expressions are well typed and omit type information whenever types can be inferred from the context. When there is no ambiguity, outer parentheses are dropped. Moreover, we adopt the convention that application associates to the left. Application and abstraction are extended to lists of terms and variables by $M\vec{N} := (\dots((MN_1)N_2)\dots)N_n$ and $\lambda\vec{x}M := \lambda x_1(\lambda x_2(\dots(\lambda x_n M)\dots))$ respectively.

NOTATION. In the sequel we will use the following notations for lists and pairs:

$$\langle\rangle := \text{nil},\ l^{\rho^*} * x^\rho := \text{cons}\, xl$$
$$\langle M^\rho, N^\sigma\rangle := \text{pair} MN\colon \rho\times\sigma.$$

Note the order in $l * x$.

3. Constructor Patterns and Computation Rules

DEFINITION 3.1. (1) *Constructor terms* are given by the grammar

$$P ::= x^\rho \mid (C\vec{P})^\mu.$$

(2) A *constructor pattern* of arity $(\sigma_1, \dots, \sigma_n)$ is a list $\vec{P} = P_1^{\sigma_1}, \dots, P_n^{\sigma_n}$ of constructor terms which is *linear*, i.e., each variable occurs at most once in $\vec{P}$. We usually denote constructor patterns by $\vec{P}, \vec{Q}$.

(3) Let $\vec{P}$ be a constructor pattern and ξ a substitution. Then ξ is called *admissible for* $\vec{P}$ if $\vec{P}\xi$ is a constructor pattern.

REMARK 3.2. (1) Let $\vec{P}$ and $\vec{Q}$ be constructor patterns of the same arity and with $\mathsf{FV}(\vec{P}) \cap \mathsf{FV}(\vec{Q}) = \emptyset$. Let ξ be a most general unifier of $\vec{P}$ and $\vec{Q}$, then ξ is admissible for $\vec{P}$ and $\vec{Q}$.

(2) Let ξ be admissible for $\vec{P}$ and $\vec{x} = \mathsf{FV}(\vec{P})$, then $\vec{x}\xi$ is again a constructor pattern.

DEFINITION 3.3. Let $D \in \mathcal{D}$ and $\triangleright^D$ a binary relation between constructor patterns and terms. Then $\triangleright^D$ is called a *system of computation rules for* D if:

(1) If $\vec{P} \triangleright^D M$, then $\mathsf{ar}(\vec{P}) = \mathsf{ar}(D)$, $\mathsf{FV}(M) \subseteq \mathsf{FV}(\vec{P})$, and M has the same type as $D\vec{P}$.

(2) If $\vec{P} \triangleright^D M$, $\vec{Q} \triangleright^D N$, and ξ is a most general unifier of $\vec{P}$ and $\vec{Q}$, then $M\xi = N\xi$ and either $\vec{P} = \vec{Q}$, or $\vec{P}$ and $\vec{Q}$ have distinct variables.

A *system of computation rules for* $\mathcal{D}$ is a family $\mathcal{P} = (\triangleright^D \mid D \in \mathcal{D})$ such that $\triangleright^D$ is a system of computation rules for each $D \in \mathcal{D}$. In this case we write $D\vec{P} \triangleright M$ for $\vec{P} \triangleright^D M$. We call $D\vec{P} \triangleright M$ a *computation rule for* D.

The restriction in (2) that $\vec{P}$ and $\vec{Q}$ should have distinct free variables whenever being different and unifiable is only of technical nature. For displaying purposes we adopt the convention that implicitly all computation rules have different free variables although we might use the same variables in their formulation. Moreover, we tacitly assume $\mathsf{FV}(\vec{P}) \subseteq \vec{y}$ whenever we write $D\vec{P}(\vec{y}) \rhd M$.

EXAMPLE. Consider the boolean connective for conjunction:

$$\mathtt{tt}\ \mathsf{and}\ y \rhd y, \quad x\ \mathsf{and}\ \mathtt{tt} \rhd x,$$
$$\mathtt{ff}\ \mathsf{and}\ y \rhd \mathtt{ff}, \quad x\ \mathsf{and}\ \mathtt{ff} \rhd \mathtt{ff}.$$

Notice that whenever two left hand sides are unifiable, the unified right hand sides are equal, e.g., in the first two rules a most general unifier sends x and y to $\mathtt{tt}$, thus the unified right hand sides are both equal to $\mathtt{tt}$. Further examples of computation rules can be found in Section 5.

For now we fix a system of computation rules for $\mathcal{D}$ denoted by $\mathcal{P}$.

4. Reduction

Computation rules inherit directly a notion of stepwise transformations of defined constants. Together with the usual β-reduction, these will constitute our reduction relation.

DEFINITION 4.1. The *reduction relation* $\longrightarrow_{\beta\mathcal{P}}$, or $\longrightarrow$ for short, on terms is inductively defined by:

$$\text{RED-BETA}\ \frac{}{(\lambda x M)N \longrightarrow M[N/x]} \qquad \text{RED-D}\ \frac{D\vec{P}(\vec{y}) \rhd M}{D\vec{P}(\vec{N}) \longrightarrow M[\vec{N}/\vec{y}]}$$

$$\text{RED-CONG-L}\ \frac{M \longrightarrow M'}{MN \longrightarrow M'N} \qquad \text{RED-CONG-R}\ \frac{N \longrightarrow N'}{MN \longrightarrow MN'}$$

$$\text{RED-XI}\ \frac{M \longrightarrow N}{\lambda x M \longrightarrow \lambda x N}$$

A term of the form $(\lambda x M)N$ is called *β-redex*. A term of the form $D\vec{P}(\vec{N})$ with $D\vec{P}(\vec{y}) \rhd M$ for some M is called *$\mathcal{P}$-redex*. If $M \longrightarrow^+ N$, then N is a *reduct* of M; if already $M \longrightarrow N$, then we say that N is an *immediate reduct* of M. A term that has no immediate reduct is called *normal*. If $M \longrightarrow^* N$ and N is normal, we call N a *normal form* of M.

Clearly, this notion of reduction does not always ensure termination. A trivial example is given by the computation rule $\mathbf{\Omega} \rhd \mathbf{\Omega}$ which yields the infinite reduction sequence $\mathbf{\Omega} \longrightarrow \mathbf{\Omega} \longrightarrow \mathbf{\Omega} \longrightarrow \dots$.

4.1. Confluence.

The aim of this subsection is to show the confluence of the reduction relation defined above. In general, the confluence property of a relation ensures that normal forms are unique whenever they exist.

DEFINITION 4.2. A binary relation $\longrightarrow_R$ has the *diamond property*, if $x \longrightarrow_R y_1$ and $x \longrightarrow_R y_2$ implies that there exists z such that $y_1 \longrightarrow_R z$ and $y_2 \longrightarrow_R z$. It is *confluent*, if $\longrightarrow_R^*$ has the diamond property.

LEMMA 4.3. *If a binary relation $\longrightarrow_R$ has the diamond property, then $\longrightarrow_R$ is confluent.*

In order to show the confluence of our reduction relation, we adapt the proof of Takahashi [**45**] which is based on the parallel reduction technique due to Tait and Martin-Löf. The parallel reduction can remove all visible redexes in one step.

In the following, B ranges over constructors and defined constants.

Definition 4.4. The *parallel reduction* is inductively defined by the clauses:

$$\textsc{par-var}\ \frac{}{x \Longrightarrow x} \qquad \textsc{par-const}\ \frac{}{B \Longrightarrow B} \qquad \textsc{par-xi}\ \frac{M \Longrightarrow M'}{\lambda x M \Longrightarrow \lambda x M'}$$

$$\textsc{par-cong}\ \frac{M \Longrightarrow M' \quad N \Longrightarrow N'}{MN \Longrightarrow M'N'} \qquad \textsc{par-beta}\ \frac{M \Longrightarrow M' \quad N \Longrightarrow N'}{(\lambda x M)N \Longrightarrow M'[N'/x]}$$

$$\textsc{par-d}\ \frac{D\vec{P}(\vec{y}) \rhd M \quad \vec{N} \Longrightarrow \vec{N}'}{D\vec{P}(\vec{N}) \Longrightarrow M[\vec{N}'/\vec{y}]}$$

Lemma 4.5. *(1) $\Longrightarrow$ is reflexive.*
(2) $\longrightarrow \subseteq \Longrightarrow \subseteq \longrightarrow^*$, hence $\Longrightarrow^* = \longrightarrow^*$.
(3) If $M \Longrightarrow M'$ and $\vec{N} \Longrightarrow \vec{N}'$, then $M[\vec{N}/\vec{x}] \Longrightarrow M'[\vec{N}'/\vec{x}]$.

Proof. (1) and (2) are easy to prove. (3) is proved by induction on $M \Longrightarrow M'$, where the only interesting case is par-d:

$$\textsc{par-d}\ \frac{D\vec{P}(\vec{y}) \rhd M \quad \vec{K} \Longrightarrow \vec{K}'}{D\vec{P}(\vec{K}) \Longrightarrow M[\vec{K}'/\vec{y}]}$$

Then $\mathsf{FV}(M) \subseteq \vec{y}$ by our restriction on computation rules and by IH we have $\vec{K}[\vec{N}/\vec{x}] \Longrightarrow \vec{K}'[\vec{N}'/\vec{x}]$. So by par-d:

$$D\vec{P}(\vec{K})[\vec{N}/\vec{x}] = D\vec{P}(\vec{K}[\vec{N}/\vec{x}]) \Longrightarrow M[\vec{K}'[\vec{N}'/\vec{x}]/\vec{y}] = (M[\vec{K}'/\vec{y}])[\vec{N}'/\vec{x}]$$

Where the last equality holds since $\mathsf{FV}(M) \subseteq \vec{y}$. □

Definition 4.6. The *complete expansion* M^* of M is defined by induction on M:

$$\begin{aligned}
x^* &:= x,\\
C^* &:= C,\\
D^* &:= D \text{ if } |\mathsf{ar}(D)| \neq 0, \text{ or } |\mathsf{ar}(D)| = 0 \text{ and } D \text{ has no rules},\\
(\lambda x M)^* &:= \lambda x M^*,\\
(M_1 M_2)^* &:= M_1^* M_2^* \text{ if } M_1 M_2 \text{ is neither a } \beta\text{- nor a } \mathcal{P}\text{-redex},\\
((\lambda x M_1) M_2)^* &:= M_1^*[M_2^*/x],\\
(D\vec{P}(\vec{N}))^* &:= M[\vec{N}^*/\vec{y}] \text{ if } D\vec{P}(\vec{y}) \rhd M.
\end{aligned}$$

Lemma 4.7. *M^* is well-defined.*

Proof. It suffices to prove that, if $D\vec{P}_1(\vec{N}_1) = D\vec{P}_2(\vec{N}_2)$ and $D\vec{P}_i(\vec{y}_i) \rhd M_i$ $(i = 1, 2)$, then $M_1[\vec{N}_1^*/\vec{y}_1] = M_2[\vec{N}_2^*/\vec{y}_2]$. Assume the premises. Then $D\vec{P}_1(\vec{N}_1) = D\vec{P}_2(\vec{N}_2)$ implies that $D\vec{P}_1(\vec{y}_1)$ and $D\vec{P}_2(\vec{y}_2)$ are unifiable, so let ξ be the most general unifier. Then $M_1\xi = M_2\xi$ by our assumption on computation rules. Moreover $\vec{N}_i = (\vec{y}_i\xi)(\vec{K}_i)$ for some $\vec{K}_i$ and w.l.o.g. we can assume that $\vec{K}_1 = \vec{K}_2$ and $\mathsf{FV}(\vec{y}_i\xi) \subseteq \vec{x}$ $(i = 1, 2)$. Now:

$$\vec{N}_i^* = ((\vec{y}_i\xi)(\vec{K}_i))^* = (\vec{y}_i\xi)(\vec{K}_i^*).$$

Where the last equality is easily seen, since $(\vec{y}_i\xi)$ is a constructor pattern. Finally, the claim follows from $M_1\xi = M_2\xi$, $\vec{K}_1^* = \vec{K}_2^*$, and:

$$M_i[\vec{N}_i^*/\vec{y}_i] = M_i[(\vec{y}_i\xi)[\vec{K}_i^*/\vec{x}]/\vec{y}_i] = (M_i[\vec{y}_i\xi/\vec{y}_i])[\vec{K}_i^*/\vec{x}] = (M_i\xi)[\vec{K}_i^*/\vec{x}]. \quad \square$$

THEOREM 4.8. *If $M \Longrightarrow N$, then $N \Longrightarrow M^*$.*

PROOF. Ind(M). Cases($M \Longrightarrow N$). The cases where M is a variable or a constructor are trivial.

Case

$$\text{PAR-CONST}\ \frac{}{D \Longrightarrow D}.$$

If $|\mathsf{ar}(D)| = 0$, or $|\mathsf{ar}(D)| \neq 0$ and D has no rules, then $D^* = D$ and we are done. Otherwise, $|\mathsf{ar}(D)| = 0$ and there is a (unique) M with $D \triangleright M$. Using PAR-D gives $D \Longrightarrow M = D^*$.

Case

$$\text{PAR-XI}\ \frac{M \Longrightarrow N}{\lambda xM \Longrightarrow \lambda xN}.$$

Then by IH $N \Longrightarrow M^*$, hence also $\lambda xN \Longrightarrow \lambda xM^* = (\lambda xM)^*$.

Case PAR-CONG. We distinguish cases on M.

Subcase $M = M_1M_2$ is neither a β- nor a $\mathcal{P}$-redex, then:

$$\text{PAR-CONG}\ \frac{M_1 \Longrightarrow N_1 \quad M_2 \Longrightarrow N_2}{M_1M_2 \Longrightarrow N_1N_2}.$$

By IH $N_1 \Longrightarrow M_1^*$ and $N_2 \Longrightarrow M_2^*$, hence $N_1N_2 \Longrightarrow M_1^*M_2^* = (M_1M_2)^*$.

Subcase $M = (\lambda xM_1)M_2 \Longrightarrow (\lambda xN_1)N_2 = N$ with $M_1 \Longrightarrow N_1$ and $M_2 \Longrightarrow N_2$. Then by IH $N_1 \Longrightarrow M_1^*$ and $N_2 \Longrightarrow M_2^*$. Hence $(\lambda xN_1)N_2 \Longrightarrow M_1^*[M_2^*/x] = ((\lambda xM_1)M_2)^*$ by PAR-BETA.

Subcase $M = D\vec{P}(\vec{M}) \Longrightarrow N$ with $D\vec{P}(\vec{y}) \triangleright K$ and $\mathsf{FV}(\vec{P}) = \vec{y}$. By arity reasons the last rules in $D\vec{P}(\vec{M}) \Longrightarrow N$ were applications of PAR-CONG and thus $N = D\vec{N}$ for some $\vec{N}$ with $\vec{P}(\vec{M}) \Longrightarrow \vec{N}$. One easily sees that $\vec{N} = \vec{P}(\vec{N}')$ with $\vec{M} \Longrightarrow \vec{N}'$. By IH, $\vec{N}' \Longrightarrow \vec{M}^*$. Moreover $(D\vec{P}(\vec{M}))^* = K[\vec{M}^*/\vec{y}]$, and hence $N = D\vec{P}(\vec{N}') \Longrightarrow K[\vec{M}^*/\vec{y}] = M^*$ by PAR-D.

Case

$$\text{PAR-BETA}\ \frac{M_1 \Longrightarrow N_1 \quad M_2 \Longrightarrow N_2}{(\lambda xM_1)M_2 \Longrightarrow N_1[N_2/x]}.$$

By IH we obtain $N_1 \Longrightarrow M_1^*$ and $N_2 \Longrightarrow M_2^*$. Now Lemma 4.5 yields $N_1[N_2/x] \Longrightarrow M_1^*[M_2^*/x] = ((\lambda xM_1)M_2)^*$.

Case

$$\text{PAR-D}\ \frac{D\vec{P}(\vec{y}) \triangleright K \quad \vec{M} \Longrightarrow \vec{N}}{D\vec{P}(\vec{M}) \Longrightarrow K[\vec{N}/\vec{y}]},$$

with $\mathsf{FV}(\vec{P}) = \vec{y}$. By IH $\vec{N} \Longrightarrow \vec{M}^*$. So by Lemma 4.5 we get $K[\vec{N}/\vec{y}] \Longrightarrow K[\vec{M}^*/\vec{y}] = (D\vec{P}(\vec{M}))^*$. $\square$

COROLLARY 4.9. *$\longrightarrow$ is confluent.*

PROOF. By Theorem 4.8 $\Longrightarrow$ satisfies the diamond property and hence is confluent by Lemma 4.3. So the claim follows since $\Longrightarrow^*$ equals $\longrightarrow^*$ by Lemma 4.5 (2). □

REMARK 4.10. If we add η-reduction, i.e., the rule

$$\frac{}{\lambda x.Mx \longrightarrow M}\ (x \notin \mathsf{FV}(M)),$$

our system is no longer confluent. Consider a defined constant D with the only computation rule $Dx \triangleright \mathtt{tt}$. Then $\lambda x.Dx \longrightarrow D$ according to the η-rule, and $\lambda x.Dx \longrightarrow \lambda x.\mathtt{tt}$. Both reducts are normal and hence don't possess a common reduct.

5. Examples: Plotkin's PCF and Gödel's T

In this section we introduce the term systems PCF and Gödel's system T. Both are instances of our theory so far.

5.1. Plotkin's PCF. The *programming language for computable functionals*, PCF for short, was first introduced by Plotkin in [**33**]. It is a programming language based on Scott's *logic for computable functionals* (LCF) [**38, 29**] and is a simplified typed functional programming language based on fixed-point operators.

We now describe the system PCF. Types are given by the type for natural numbers $\mathbf{N}$ and if ρ, σ are types, then so is $\rho \to \sigma$. The defined constants are $\mathsf{pred}^{\mathbf{N}\to\mathbf{N}}, \mathsf{ifz}^{\mathbf{N}\to\mathbf{N}\to\mathbf{N}\to\mathbf{N}}$, and for every type ρ, $\mathcal{Y}_\rho$ of type $(\rho \to \rho) \to \rho$. As computation rules we take the following:

$$\begin{array}{cc} \mathsf{pred}\,0 \triangleright 0 & \mathsf{ifz}\,0yz \triangleright y \\ \mathsf{pred}(Sx) \triangleright x & \mathsf{ifz}(Sx)yz \triangleright z \\ \multicolumn{2}{c}{\mathcal{Y}_\rho f \triangleright f(\mathcal{Y}_\rho f)} \end{array} \tag{1}$$

The resulting term system is called PCF.

REMARK 5.1. The system above is not exactly the same as Plotkin defined it in [**33**]. There, additionally to $\mathbf{N}$, the booleans $\mathbf{B}$ are taken as a base type. Besides constants for the booleans and some minor differences in the choice of the constants, Plotkin uses a constant k_n for each $n \in \mathbb{N}$ instead of constructors and formulates the reduction rules using these numerals, e.g., his predecessor constant is defined on numerals k_n only. This can as well be emulated in our system using $\mathsf{pred}(S^{n+1}0) \triangleright S^n 0$ as a computation rule for each $n \in \mathbb{N}$. (Recall that we allow infinitely many computation rules.) In fact, this results in a different reduction relation: If we define $\infty := \mathcal{Y}_{\mathbf{N}}(S)$, then $\mathsf{pred}\,\infty$ only reduces to terms of the form $\mathsf{pred}(S^n\infty)$ in Plotkin's formulation, whereas in our formulation it also reduces to ∞. However, with our intended semantics, i.e., a non-flat semantic domain for $\mathbf{N}$ (see the next chapter), the formulation in (1) seems to be more natural.

5.2. Gödel's T. The system T is a generalization of the primitive recursive functions to higher (finite) types and was introduced by Gödel in [**18**]. The essential ingredient of the system is the constant for primitive (or structural) recursion in higher types which in fact was already introduced by Hilbert in [**20**]. Let us fix an arbitrary system of types $\mathcal{B}$.

Structural Recursion. Let $\vec{\mu} = \mu\vec{\alpha}.\vec{\kappa}$. For the i-th constructor type

$$\kappa_i = \vec{\rho} \to (\vec{\sigma}_1 \to \alpha_{j_1}) \to \cdots \to (\vec{\sigma}_n \to \alpha_{j_n}) \to \alpha_j \tag{2}$$

and given types $\vec{\tau}$ with $|\vec{\tau}| = |\vec{\mu}|$, we define the i-th *step type*

$$\begin{aligned}\delta_i^{\vec{\mu},\vec{\tau}} := \vec{\rho} &\to (\vec{\sigma}_1 \to \mu_{j_1}) \to \cdots \to (\vec{\sigma}_n \to \mu_{j_n}) \\ &\to (\vec{\sigma}_1 \to \tau_{j_1}) \to \cdots \to (\vec{\sigma}_n \to \tau_{j_n}) \to \tau_j.\end{aligned}$$

Then the j-th *recursion operator* $\mathcal{R}_{\mu_j}^{\vec{\mu},\vec{\tau}}$ is given as a defined constant of type

$$\mathcal{R}_{\mu_j}^{\vec{\mu},\vec{\tau}} \colon \mu_j \to \delta_1^{\vec{\mu},\vec{\tau}} \to \cdots \to \delta_k^{\vec{\mu},\vec{\tau}} \to \tau_j,$$

where k is the length of $\vec{\kappa}$. If $\vec{\mu}$ and $\vec{\tau}$ are clear from the context, we also write $\mathcal{R}_j$ or $\mathcal{R}_j^{\vec{\tau}}$ for $\mathcal{R}_{\mu_j}^{\vec{\mu},\vec{\tau}}$ and in the case of non-simultaneously defined base types we drop the subscript. For each constructor $C_i^{\vec{\mu}} \colon \kappa_i[\vec{\mu}/\vec{\alpha}]$ with κ_i as in (2) we add the following computation rule:

$$\mathcal{R}_j(C_i^{\vec{\mu}}\vec{x})\vec{y} \rhd y_i\vec{x}\big(\lambda\vec{z}_1^{\vec{\sigma}_1}.\mathcal{R}_{j_1}(x_1^R\vec{z}_1)\vec{y}\big)\ldots\big(\lambda\vec{z}_n^{\vec{\sigma}_n}.\mathcal{R}_{j_n}(x_n^R\vec{z}_n)\vec{y}\big)$$

where y_i is of type $\delta_i^{\vec{\mu},\vec{\tau}}$ and $\vec{x} = \vec{x}^P, \vec{x}^R$ with $\vec{x}^P \colon \vec{\rho}$ (the parameter arguments) and $x_i^R \colon \vec{\sigma}_i \to \mu_{j_i}$ (the recursive arguments).

Examples. (1) For the type of natural numbers $\mathbf{N}$ this amounts to the usual equations for primitive recursion in higher types. The type of the recursion operator $\mathcal{R} := \mathcal{R}_{\mathbf{N}}^{\mathbf{N},\tau}$ is

$$\mathcal{R} \colon \mathbf{N} \to \tau \to (\mathbf{N} \to \tau \to \tau) \to \tau$$

and the computation rules are

$$\mathcal{R}0xf \rhd x \quad \text{and} \quad \mathcal{R}(Sn)xf \rhd fn(\mathcal{R}nxf).$$

(2) For pair types $\rho \times \sigma$ we can define the left and right projections, π_0 and π_1, using the recursion operators $\mathcal{R}^\rho := \mathcal{R}_{\rho\times\sigma}^{\rho\times\sigma,\rho}$ and $\mathcal{R}^\sigma := \mathcal{R}_{\rho\times\sigma}^{\rho\times\sigma,\sigma}$ respectively:

$$\begin{aligned}\pi_0 &:= \lambda p^{\rho\times\sigma}.\mathcal{R}^\rho p(\lambda x^\rho \lambda y^\sigma.x),\\ \pi_1 &:= \lambda p^{\rho\times\sigma}.\mathcal{R}^\sigma p(\lambda x^\rho \lambda y^\sigma.y).\end{aligned}$$

Notice that $\pi_0(\langle M, N\rangle) \longrightarrow^* M$ and $\pi_1(\langle M, N\rangle) \longrightarrow^* N$, but in general, $\langle \pi_0(K), \pi_1(K)\rangle$ does not reduce to K.

The System $T_{\mathcal{B}}$. Gödel's system T for a system of types $\mathcal{B}$ is denoted by $T_{\mathcal{B}}$. The terms of $T_{\mathcal{B}}$ are all $(\mathcal{B}, \mathcal{D})$-terms, where $\mathcal{D}$ consists of all defined constants for structural recursion. Reduction on these terms is defined by the $\longrightarrow_{\beta\mathcal{P}}$-reduction, where $\mathcal{P}$ is given by the computation rules for structural recursion given above.

It is well-known that this reduction relation is strongly normalizing, i.e., every reduction sequence is finite (see [**36**, Corollary 6.13] for a proof in our setting). It follows by confluence (cf. Section 4.1) that each term possesses a unique normal form.

6. Heyting and Peano Arithmetic

In this section we introduce intuitionistic and classical arithmetic over an arbitrary system of types $\mathcal{B}$ containing $\mathbf{N}$. Our presentation is mainly based on [**47**] and [**35**]. This section, however, only serves the purpose of fixing our systems instead of giving a comprehensive introduction to the subject.

It is convenient to allow arbitrary predicate parameters in our systems. Let $\mathcal{X}$ be a set of predicate variables where each $X \in \mathcal{X}$ comes with a fixed arity $\mathsf{ar}(X)$. Moreover, let $\bot$ be a nullary predicate variable denoting falsity, and for each type ρ, let $=_\rho$ be a binary predicate symbol of arity (ρ, ρ).

DEFINITION 6.1. The formulas of $\mathsf{HA}^\omega[\mathcal{X}]$ are given by

$$A, B ::= M^\rho =_\rho N^\rho \mid \bot \mid X(\vec{M}^{\vec{\rho}}) \mid A \wedge B \mid A \vee B \mid A \to B \mid \forall x^\rho A \mid \exists x^\rho A,$$

where $\vec{M}^{\vec{\rho}}$,M^ρ, and N^ρ are terms of Gödel's $T_\mathcal{B}$, and $X \in \mathcal{X}$ has arity $(\vec{\rho})$. If $\mathcal{X}$ is empty, we simply write HA^ω instead of $\mathsf{HA}^\omega[\emptyset]$.

Negation is defined by $\neg A := A \to \bot$.

A $\mathsf{HA}^\omega[\mathcal{X}]$-*predicate* is a $\mathsf{HA}^\omega[\mathcal{X}]$-formula A with distinguished variables $\vec{x}^{\vec{\sigma}}$, in symbols $\lambda\vec{x}^{\vec{\sigma}}.A$. The arity of the a $\mathsf{HA}^\omega[\mathcal{X}]$-predicate is defined as $\mathsf{ar}(\lambda\vec{x}^{\sigma}.A) := (\vec{\sigma})$. We set $(\lambda\vec{x}^{\vec{\sigma}}.A)(\vec{t}^{\vec{\sigma}}) := A[\vec{t}/\vec{x}]$. Usually we are more informal concerning predicates: If a formula A has been introduced with distinguished variables $\vec{x}$ as $A(\vec{x})$, we may also view this formula as the predicate $\lambda\vec{x}.A$.

Let Γ, Δ range over sets of $\mathsf{HA}^\omega[\mathcal{X}]$-formulas. As usual, we write Γ, Δ or $\Gamma + \Delta$ for $\Gamma \cup \Delta$ and use similar notations for formulas.

6.1. Axioms and Logic. We adopt the convention that all axioms are implicitly universally closed.

6.1.1. *Induction.* Let simultaneously defined algebras $\vec{\mu} = \mu\vec{\alpha}.\vec{\kappa}$ and goal formulas $\forall x_j^{\mu_j} A_j(x_j)$ of $\mathsf{HA}^\omega[\mathcal{X}]$ be given. For the i-th constructor type $\kappa_i \in \vec{\kappa}$ of the form

$$\kappa_i = \vec{\rho} \to (\vec{\sigma}_1 \to \alpha_{j_1}) \to \cdots \to (\vec{\sigma}_n \to \alpha_{j_n}) \to \alpha_j$$

we define the i-th *step formula* D_i as

$$\begin{aligned} D_i := \forall \vec{y}^{\vec{\rho}}, z_1^{\vec{\sigma}_1 \to \mu_{j_1}}, \dots, z_n^{\vec{\sigma}_n \to \mu_{j_n}}. \\ \forall \vec{x}_1^{\vec{\sigma}_1} A_{j_1}(z_1\vec{x}_1) \to \dots \to \forall \vec{x}_n^{\vec{\sigma}_n} A_{j_n}(z_n\vec{x}_n) \to A_j(C_i\vec{y}\vec{z}), \end{aligned}$$

where C_i is the i-th constructor of $\vec{\mu}$.

The *induction axioms* of $\mathsf{HA}^\omega[\mathcal{X}]$ are given by

$$\mathrm{Ind}_{\mu_j}^{\vec{x},\vec{A}}: \quad \forall x_j \big(D_1 \to \dots \to D_k \to A_j(x_j)\big),$$

where the $A_j(x_j)$'s range over $\mathsf{HA}^\omega[\mathcal{X}]$-formulas.

EXAMPLES. As special cases of the general form, we state the induction axioms for $\mathbf{N}$ and pair types $\rho \times \sigma$:

$$\begin{aligned} \mathrm{Ind}_{\mathbf{N}}^{n,A}: &\quad \forall n\big(A(0) \to \forall n(A(n) \to A(Sn)) \to A(n)\big), \\ \mathrm{Ind}_{\rho\times\sigma}^{z,A}: &\quad \forall z^{\rho\times\sigma}\big(\forall x^\rho, y^\sigma A(\langle x, y\rangle) \to A(z)\big). \end{aligned}$$

6.1.2. *Equality Axioms.* To simplify equational reasoning we postulate equality between terms with the same normal form as follows. For any two terms M^ρ and N^ρ with the same normal form, we add the axiom

$$M^\rho =_\rho N^\rho.$$

In particular, this includes reflexivity $x =_\rho x$. The *compatibility axiom* is given by

$$x =_\rho y \to A[x/z] \to A[y/z],$$

where A ranges over $\mathsf{HA}^\omega[\mathcal{X}]$-formulas.

6.1.3. *Logic.* As base logical system we use (many sorted) minimal logic based on natural deduction. We use natural deduction with assumptions labeled by assumption variables as presented, e.g., in [**48**].

We write $\Gamma \vdash_{\mathrm{m}} A$ if there is a derivation of A from Γ in minimal logic, and $\Gamma \vdash_{\mathrm{i}} A$ if there is a derivation in minimal logic using *ex falso quodlibet*, i.e., $\Gamma + \mathrm{Efq} \vdash_{\mathrm{m}} A$, where Efq is given by the axiom scheme

$$\mathrm{Efq}_B\colon \qquad \bot \to B,$$

for B an $\mathsf{HA}^\omega[\mathcal{X}]$-formula. Analogously, classical derivability $\Gamma \vdash_{\mathrm{c}} A$ is defined as $\Gamma + \mathrm{Stab} \vdash_{\mathrm{m}} A$, where *stability* Stab is the axiom scheme

$$\mathrm{Stab}_B\colon \qquad \neg\neg B \to B,$$

where B ranges over $\mathsf{HA}^\omega[\mathcal{X}]$-formulas.

6.2. Description of the axiom system $\mathsf{HA}^\omega[\mathcal{X}]$. The axioms of *Heyting Arithmetic in higher types*, $\mathsf{HA}^\omega[\mathcal{X}]$, are the induction axioms, equality axioms, and the axiom (with $F := 0 =_{\mathbf{N}} 1$)

$$F \to X(\vec{z})$$

for each $X \in \mathcal{X} \cup \{\bot\}$. We identify the system $\mathsf{HA}^\omega[\mathcal{X}]$ with its axioms and hence we write $\mathsf{HA}^\omega[\mathcal{X}] + \Gamma \vdash_* A$, $* = \mathrm{m}, \mathrm{i}, \mathrm{c}$ to mean that there is a derivation of A in minimal-, intuitionistic-, and classical logic respectively from the axioms of $\mathsf{HA}^\omega[\mathcal{X}]$ and Γ. We also write $\mathsf{HA}^\omega[\mathcal{X}] + \Gamma \vdash A$, or say that A *is provable in* $\mathsf{HA}^\omega[\mathcal{X}]$ *from* Γ instead of $\mathsf{HA}^\omega[\mathcal{X}] + \Gamma \vdash_{\mathrm{i}} A$.

Peano Arithmetic in higher types $\mathsf{PA}^\omega[\mathcal{X}]$ is defined as $\mathsf{HA}^\omega[\mathcal{X}]$ but based on classical logic. Thus we write $\mathsf{PA}^\omega[\mathcal{X}] + \Gamma \vdash A$, or say that A *is provable in* $\mathsf{PA}^\omega[\mathcal{X}]$ *from* Γ instead of $\mathsf{PA}^\omega[\mathcal{X}] + \Gamma \vdash_{\mathrm{c}} A$. Of course, we could have just used classical derivability and $\mathsf{HA}^\omega[\mathcal{X}]$ instead of introducing $\mathsf{PA}^\omega[\mathcal{X}]$; we prefer the latter to emphasize classical derivability.

To be more precise we should have written $\mathsf{HA}^\omega_{\mathcal{B}}[\mathcal{X}]$ to make the reference to the system of types $\mathcal{B}$ explicit. We have not done so because later we will focus on HA^ω for a fixed system of types.

Note. Instead of having an additional predicate for logical falsity $\bot$ we could have used arithmetical falsity $F = (0 =_{\mathbf{N}} 1)$ and based HA^ω on minimal logic. Having $\bot$ in the language, however, allows a clearer approach to A-translation (cf. Chapter 4).

We conclude this subsection by listing some properties derivable in the system $\mathsf{HA}^\omega[\mathcal{X}]$.

Since terms with the same normal form are equal (in HA^ω), it is provable (in minimal logic) that constructors are injective (e.g., for the successor

$S^{\mathbf{N}\to\mathbf{N}}$ use compatibility together with the predecessor function which is definable using primitive recursion).

By induction on the buildup of $\mathsf{HA}^\omega[\mathcal{X}]$-formulas, one easily sees that $F \to A$ is derivable from $\mathsf{HA}^\omega[\mathcal{X}]$ in minimal logic. Because we have $\neg 0 =_{\mathbf{N}} 1$ (i.e., $F \to \bot$), one can prove $\forall n^{\mathbf{N}}(n = 0 \vee n \neq 0)$ by induction. Here $n \neq 0$ means $\neg(n = 0)$.

6.3. Extensionality. The system $\mathsf{E{-}HA}^\omega[\mathcal{X}]$ of extensional Heyting Arithmetic is like $\mathsf{HA}^\omega[\mathcal{X}]$ with the additional axiom (scheme) of extensionality Ext given by:

$$\mathrm{Ext}_{\rho,\sigma} \qquad \forall x^\rho\, yx =_\sigma zx \to y =_{\rho\to\sigma} z,$$

where ρ, σ range over $\mathcal{B}$. Extensional Peano Arithmetic $\mathsf{E{-}PA}^\omega[\mathcal{X}]$ is defined analogously.

CHAPTER 2

Semantics

In this chapter we develop the intended semantics of the syntactical concepts developed in the last chapter. This guarantees that our recursion schemes in Chapter 3 are consistent.

The underlying concept of our semantics is that of an information system introduced by Scott [**39**]. Information systems directly incorporate the concept of a "finite approximation" through so called *formal neighborhoods*; the objects of our semantic domains are the ideals – possibly infinite objects – of the information systems.

In the first section we define coherent information systems and study their abstract properties. Section 2 introduces the information systems $\mathbf{C}_\rho$ and the partial continuous functionals as their ideals. In Sections 3–5 we introduce the denotation of a term as a partial continuous functional and investigate the relationship between the reduction relation on terms and the denotation of terms, i.e., their operational and denotational behavior. Sections 6 and 7 of the present chapter study the total among the partial continuous functionals.

1. Coherent Information Systems

Information systems were first introduced by Scott in [**39**] as an alternative way of presenting Scott domains. They provide an axiomatic approach, similar to logical deductive systems, to describe approximations of ideal objects by "finite" ones.

Often, the approach to higher type computation is presented using abstract domain theory; in a Scott domain each element is the supremum of certain compact (or finite) elements, or put in other words, each element is approximated by finite ones. Here we take a different approach: we begin by describing the finite elements.

It turns out that we only need particular information systems, namely the coherent ones. We take those as our fundamental structure because they allow a particularly terse presentation.

Definition 1.1. A *coherent information system* (abbreviated *c.i.s.*) is a triple $\mathbf{A} = (A, \smile, \vdash)$ such that:

(1) A is a countable non-empty set.
(2) $\smile$ is a reflexive and symmetric binary relation on A (called the *consistency relation*).
(3) $\vdash$ is a binary relation between $\mathrm{Con}_\mathbf{A}$ and A (called the *entailment relation*), where

$$\mathrm{Con}_\mathbf{A} := \{U \mid U \subseteq^{fin} A \wedge \forall a, b \in U (a \smile b)\}$$

is the set of all *formal neighborhoods* of $\mathbf{A}$, with the following properties for all $U, V \in \mathrm{Con}_{\mathbf{A}}$ and $a, b \in A$:
(a) $a \in U \to U \vdash a$,
(b) $\forall b \in V (U \vdash b) \wedge V \vdash a \to U \vdash a$,
(c) $a \in U \wedge U \vdash b \to a \smile b$.

The elements of A are called *tokens* (or *atoms*) and are denoted by $a, b, c, \dots$, formal neighborhoods by $U, V, W, \dots$.

Definition 1.2. Entailment and consistency are extended to formal neighborhoods in the following way:
(1) $U \vdash V :\leftrightarrow \forall b \in V (U \vdash b)$
(2) $U \smile V :\leftrightarrow U \cup V \in \mathrm{Con}$
(3) $U \smile a :\leftrightarrow a \smile U :\leftrightarrow \{a\} \smile U$

Lemma 1.3. *(1)* $U \vdash U$
(2) $U \vdash V \wedge V \vdash W \to U \vdash W$
(3) $U \vdash V \to U \smile V$
(4) $U' \supseteq U \wedge U \vdash V \wedge V \supseteq V' \to U' \vdash V'$
(5) $U \vdash V_1, V_2 \to V_1 \smile V_2$

Proof. (1)–(4) are easy to see. For (5) assume that $U \vdash V_1, V_2$ and let $a_i \in V_i$ for $i = 1, 2$. We have to show $a_1 \smile a_2$. By assumption $U \vdash a_1$, so $U \cup \{a_1\}$ is consistent (and hence in Con) by (3). From (4) we get $U \cup \{a_1\} \vdash a_2$ and therefore $a_1 \smile a_2$. □

Note that if we define $U \leq V$ as $V \vdash U$, then $\leq$ is a preorder on Con with least element $\emptyset$. Moreover, by (5), two elements with an upper bound have a least upper bound (namely their union).

Definition 1.4. Let $\mathbf{A} = (A, \smile, \vdash)$ be a coherent information system and $x \subseteq A$. Then x is *consistent* if $a \smile b$ for all $a, b \in x$. Moreover x is *deductively closed* if for all $U \in \mathrm{Con}$ with $U \subseteq x$ and $a \in A$, $U \vdash a$ implies $a \in x$. The *deductive closure* $\overline{x}$ of x is defined as

$$\overline{x} := \{a \mid \exists U \in \mathrm{Con} . U \subseteq x \wedge U \vdash a\}.$$

Lemma 1.5. *Let* $\mathbf{A} = (A, \smile, \vdash)$ *be a c.i.s. and* $x \subseteq A$. *Then* $\overline{x}$ *is deductively closed.*

Proof. Let $U \subseteq \overline{x}$ and $U \vdash a$; we must show $a \in \overline{x}$. For each $b \in U$ there is a $V_b \subseteq x$ with $V_b \vdash b$. Thus also $V_b \smile b \smile U$, and hence $W := \bigcup_{b \in U} V_b \in \mathrm{Con}_{\mathbf{A}}$. Clearly, $x \supseteq W \vdash U \vdash a$, and so $a \in \overline{x}$ as required. □

Definition 1.6. Let $\mathbf{A} = (A, \smile_A, \vdash_A)$ and $\mathbf{B} = (B, \smile_B, \vdash_B)$ be two coherent information systems. Then the *exponent* $\mathbf{B}^{\mathbf{A}}$ is defined as the triple $(C, \vdash, \smile)$, where:
(1) $C := \mathrm{Con}_{\mathbf{A}} \times B$
(2) $(U, a) \smile (V, b) :\leftrightarrow (U \smile_A V \to a \smile_B b)$
(3) $W \vdash (V, b) :\leftrightarrow W \in \mathrm{Con} \wedge W(V) \vdash b$

where *application* for formal neighborhoods $W = \{(U_1, a_1), \dots, (U_n, a_n)\}$ is defined by

$$W(U) := \{a_i \mid U \vdash_A U_i\}.$$

LEMMA 1.7. *Application for neighborhoods is monotone:*
(1) $U \vdash U' \to W(U) \supseteq W(U')$,
(2) $W \vdash W' \to W(U) \vdash W'(U)$.

PROOF. (1) is easy to see. For (2) let $a \in W'(U)$, then there exists V with $(V, a) \in W'$ and $U \vdash V$. By assumption $W \vdash (V, a)$, i.e., $W(V) \vdash a$ and therefore $W(U) \supseteq W(V) \vdash a$. □

LEMMA 1.8. *If* $\mathbf{A}$ *and* $\mathbf{B}$ *are coherent information systems, then so is* $\mathbf{B}^{\mathbf{A}}$.

PROOF. It is clear that the consistency relation is reflexive and symmetric. We now check the rest of the axioms:

(1) Let $(U, a) \in W$, then clearly $a \in W(U)$, so $W(U) \vdash a$, i.e., $W \vdash (U, a)$.
(2) Let $W \vdash W'$ and $W' \vdash (U, a)$. We have to show $W \vdash (U, a)$. But $W' \vdash (U, a)$ implies $W'(U) \vdash a$ so $W(U) \vdash a$ by Lemma 1.7.
(3) Let $(U, a) \in W$ and $W \vdash (V, b)$. We have to show that $(U, a) \smile (V, b)$. So assume $U \cup V \in \mathrm{Con}_{\mathbf{A}}$. We show: $a \smile_B b$. We have $U \cup V \vdash_A U, V$, hence by Lemma 1.7 $W(U \cup V) \supseteq W(U), W(V)$. Furthermore $W(U) \vdash_A a$ and $W(V) \vdash_A b$, so $W(U \cup V) \vdash_A a, b$, and hence $a \smile_B b$. □

DEFINITION 1.9. Let $\mathbf{A}$ and $\mathbf{B}$ be coherent information systems. A relation $r \subseteq \mathrm{Con}_{\mathbf{A}} \times B$ is an *approximable map* from $\mathbf{A}$ to $\mathbf{B}$, if:

(1) $r(U, a) \wedge r(U, b) \to a \smile_B b$, and
(2) $W \vdash_A V \wedge r(V, U) \wedge U \vdash_B a \to r(W, a)$,

where we have written $r(V, U)$ for $\forall b \in U\, r(V, b)$. The set of all approximable maps from $\mathbf{A}$ to $\mathbf{B}$ is denoted by $\mathrm{Appr}(\mathbf{A}, \mathbf{B})$.

DEFINITION 1.10. Let $\mathbf{A}$ be a coherent information system and $x \subseteq A$. Then x is an *ideal* of $\mathbf{A}$ if x is consistent and deductively closed. The set of all ideals of $\mathbf{A}$ is denoted by $|\mathbf{A}|$. Ideals will usually be denoted by x, y, z.

LEMMA 1.11. *If* $\mathbf{A}$ *and* $\mathbf{B}$ *are coherent information systems, then*

$$\mathrm{Appr}(\mathbf{A}, \mathbf{B}) = |\mathbf{B}^{\mathbf{A}}|.$$

PROOF. Let $r \in \mathrm{Appr}(\mathbf{A}, \mathbf{B})$. We first show that r is consistent: let $(U, a), (V, b) \in r$ with $U \cup V \in \mathrm{Con}_{\mathbf{A}}$. By (2) and $U \cup V \vdash U, V$ we have $r(U \cup V, a)$ and $r(U \cup V, b)$ hence $a \smile b$ by (1).

Next, we show that r is deductively closed. Let $W \subseteq r$ and $W \vdash (V, b)$, i.e., $W(V) \vdash b$. We need $r(V, b)$. By (2) it suffices to show $r(V, W(V))$, so let $a \in W(V)$; then there exists U with $(U, a) \in W$ and $V \vdash U$. Since $W \subseteq r$ we have $r(U, a)$, so $r(V, a)$ by (2). This concludes the proof that r is an ideal.

The converse direction is easy. □

DEFINITION 1.12. Let $\mathbf{A}$ be an c.i.s. and $U \in \mathrm{Con}_{\mathbf{A}}$, then the *cone* $\widetilde{U}$ over U is $\widetilde{U} = \{x \in |\mathbf{A}| \mid x \supseteq U\}$.

Note that for formal neighborhoods U, V, we have $\widetilde{U} \cap \widetilde{V} = \widetilde{(U \cup V)}$. So the cones form a basis of a topology on $|\mathbf{A}|$, the so called *Scott topology* on $|\mathbf{A}|$.

LEMMA 1.13. *Let* $\mathbf{A}$ *be a c.i.s. and* $B \subseteq |\mathbf{A}|$. *Then the following are equivalent:*

(1) B *is open w.r.t. the Scott topology,*
(2) (a) B *is monotone, i.e.,*

$$x \in B \to x \subseteq y \to y \in B,$$

(b) B *satisfies the Scott condition, i.e.,*

$$x \in B \to \exists U \subseteq x\,(\overline{U} \in B),$$

(3) $x \in B \leftrightarrow \exists U \subseteq x\,(\overline{U} \in B)$.

PROOF. (1) $\to$ (2). Let $x \in B$ and $x \subseteq y$. Then, since B is open, there exists a neighborhood U with $\widetilde{U} \subseteq B$ and $x \in \widetilde{U}$, so $U \subseteq x \subseteq y$, i.e., $y \in \widetilde{U} \subseteq B$. Note that $\overline{U} \in \widetilde{U}$, hence $\overline{U} \in B$.

(2) $\to$ (3). Obvious.

(3) $\to$ (1). For $x \in B$ exists a $U \subseteq x$ such that $\overline{U} \in B$, so $x \in \widetilde{U}$, and $\widetilde{U} \subseteq B$ since $U \subseteq z$ implies $z \in B$ by assumption. Therefore B is open. □

DEFINITION 1.14. Let $\mathbf{A}$ and $\mathbf{B}$ be coherent information systems. A subset $D \subseteq |\mathbf{A}|$ is called *directed* if every finite subset D_0 of D has an upper bound w.r.t. $\subseteq$ in D,[1] i.e.,

$$\forall D_o \subseteq^{fin} D \exists x \in D \forall y \in D_0\, y \subseteq x.$$

A map $f\colon |\mathbf{A}| \to |\mathbf{B}|$ is *monotone* if $f(x) \subseteq f(y)$ whenever $x \subseteq y$.

LEMMA 1.15. *Let* $\mathbf{A}$ *be a coherent information system. Then:*

(1) If $D \subseteq |\mathbf{A}|$ *is directed, then* $\bigcup D \in |\mathbf{A}|$ *and* $\bigcup D$ *is the supremum (w.r.t.* $\subseteq$*) of* D.
(2) If $B \subseteq |\mathbf{A}|$ *is bounded (i.e., there is an* $x \in |\mathbf{A}|$ *such that for all* $z \in B$, $z \subseteq x$*), then* $\overline{\bigcup B} \in |\mathbf{A}|$ *and* $\overline{\bigcup B}$ *is the supremum of* B.

PROOF. Easy. □

REMARK 1.16. Without giving the proofs or notions we note that for a c.i.s. $\mathbf{A}$, $(|\mathbf{A}|, \subseteq, \emptyset)$ is an algebraic, directed and bounded complete partial order with least element $\emptyset$ — or, in other words, a *Scott-Ershov domain*. We refer the reader to [**43**] for more details. Moreover, $(|\mathbf{A}|, \subseteq, \emptyset)$ possesses the property that each subset B of $|\mathbf{A}|$ with $x \cup y$ consistent for all $x, y \in B$, has a least upper bound in $|\mathbf{A}|$.

LEMMA 1.17. *Let* $\mathbf{A}$ *and* $\mathbf{B}$ *be c.i.s. and* $f\colon |\mathbf{A}| \to |\mathbf{B}|$ *be a map. Then the following are equivalent:*

(1) f *is continuous w.r.t. the Scott topology,*
(2) f *is monotone and has the principle of finite support, i.e.,*

$$b \in f(x) \to \exists U \subseteq x(b \in f(\overline{U})),$$

(3) f *is monotone and for all* $D \subseteq |\mathbf{A}|$ *directed,*

$$f(\bigcup D) = \bigcup_{x \in D} f(x).$$

[1]Notice that directed sets are non-empty.

Proof. (1) $\to$ (2). Suppose f is continuous and $x \subseteq y$. We have to show that $f(x) \subseteq f(y)$, so let $a \in f(x)$, i.e., $f(x) \in \widetilde{\{a\}}$. Then $x \in f^{-1}(\widetilde{\{a\}})$ and since f is continuous and $\widetilde{\{a\}}$ is open, there exists U with $x \in \widetilde{U} \subseteq f^{-1}(\widetilde{\{a\}})$. Now $U \subseteq x \subseteq y$, so $y \in \widetilde{U} \subseteq f^{-1}(\widetilde{\{a\}})$, hence $a \in f(y)$.

(2) $\to$ (3). Let $D \subseteq |\mathbf{A}|$ be directed. Since f is monotone it suffices to prove

$$f(\bigcup D) \subseteq \bigcup_{x \in D} f(x).$$

So let $a \in f(\bigcup D)$; then by the principle of finite support there exists $U \subseteq \bigcup D$ with $a \in f(\overline{U})$. Since D is directed and U is finite there exists $x \in D$ with $U \subseteq x$. So $a \in f(\overline{U}) \subseteq f(x) \subseteq \bigcup_{x \in D} f(x)$, where the first inclusion holds by the monotonicity of f.

(3) $\to$ (1). Let U be a formal neighborhood of $\mathbf{B}$. It suffices to show that $f^{-1}(\widetilde{U})$ is open. Let $x \in f^{-1}(\widetilde{U})$ and $D_x := \{\overline{V} \mid V \subseteq x\}$. Clearly, D_x is directed and $x = \bigcup D_x$, and therefore by assumption we obtain $U \subseteq f(x) = \bigcup_{V \subseteq x} f(\overline{V})$. By monotonicity also $f(D_x) = \{f(\overline{V}) \mid V \subseteq x\}$ is directed, so $U \subseteq f(\overline{V})$ for some $V \subseteq x$. Thus $\overline{V} \in f^{-1}(\widetilde{U})$, yielding $x \in \widetilde{V} \subseteq f^{-1}(\widetilde{U})$. □

Definition 1.18. Let $f\colon |\mathbf{A}| \to |\mathbf{B}|$ be continuous w.r.t. the Scott topology. Then the ideal $\widehat{f} \in |\mathbf{B}^{\mathbf{A}}|$ is defined by

$$\widehat{f}(U, b) :\leftrightarrow b \in f(\overline{U}).$$

Let $r \in |\mathbf{B}^{\mathbf{A}}|$ be an ideal; then we can associate a continuous map given by

$$|r|(z) := \{b \in B \mid \exists U \subseteq^{fin} z. r(U, b)\}.$$

Lemma 1.19. *Let $f\colon |\mathbf{A}| \to |\mathbf{B}|$ be continuous and $r \in |\mathbf{B}^{\mathbf{A}}|$. Then $\widehat{f}$ and $|r|$ are well defined. Moreover, $f = |\widehat{f}|$ and $r = \widehat{|r|}$.*

Notation. By virtue of the last lemma, we sometimes write f instead of $\widehat{f}$ and r instead of $|r|$.

2. Partial Continuous Functionals

In this section we will define a coherent information system $\mathbf{C}_\rho$ for each type ρ. For this, we first introduce some notations:

Definition 2.1. An *extended token* is either a token or new distinguished symbol $*$. Extended tokens are denoted by $a^*, b^*, c^*, \ldots$. We extend the entailment and consistency relation to extended tokens by:

$$\begin{aligned} a^* \smile b^* &:\leftrightarrow a^* \text{ or } b^* \text{ are } *, \text{ or otherwise } a^* \smile b^*, \\ U \cup \{*\} \vdash a^* &:\leftrightarrow U \vdash a^*, \text{ and} \\ U \vdash * &\text{ is defined to be true.} \end{aligned}$$

Definition 2.2. The coherent information systems $\mathbf{C}_\rho = (C_\rho, \smile_\rho, \vdash_\rho)$ ($\rho \in \mathsf{Ty}$) are defined as follows. The tokens and the consistency relation are defined inductively on the syntactic complexity by the rules:

$$\frac{U \in \mathrm{Con}_\rho \quad a \in C_\sigma}{(U, a) \in C_{\rho \to \sigma}} \qquad \frac{U \smile_\rho V \to a \smile_\sigma b}{(U, a) \smile_{\rho \to \sigma} (V, b)} \qquad \frac{\vec{b}^* \in C_{\vec{\tau}} \cup \{*\}}{C\vec{b}^* \in C_\mu} \qquad \frac{\vec{a}^* \smile_{\vec{\tau}} \vec{b}^*}{C\vec{a}^* \smile_\mu C\vec{b}^*}$$

where in the last two rules C is a constructor of type $\vec{\tau} \to \mu$. Here $U \in \mathrm{Con}_\rho$ is an abbreviation for the premises that U is finite and for all $a, b \in U$ $a \smile_\rho b$. Moreover, in the second rule we have left the premises $U, V \in \mathrm{Con}_\rho$ and $a, b \in C_\sigma$ implicit in our notation. The entailment relation $\vdash_\rho$ between formal neighborhoods and tokens of type ρ is inductively defined by the rules:

$$\frac{W(V) \vdash_\sigma b}{W \vdash_{\rho\to\sigma} (V, b)} \qquad \frac{\{a_{11}^*, \dots, a_{n1}^*\} \vdash_{\tau_1} b_1^* \quad \dots \quad \{a_{1k}^*, \dots, a_{nk}^*\} \vdash_{\tau_k} b_k^*}{\{C\vec{a}_1^*, \dots, C\vec{a}_n^*\} \vdash_\mu C\vec{b}^*}\ (n \geq 1)$$

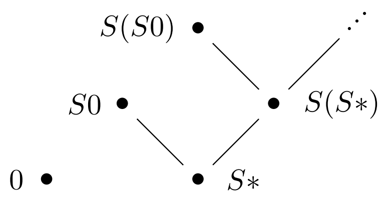
where in the last rule C is a constructor of type $\vec{\tau} \to \mu$ and $\vec{\tau} = \tau_1, \dots, \tau_k$. Let Con_ρ denote $\mathrm{Con}_{\mathbf{C}_\rho}$.

Lemma 2.3. *$\mathbf{C}_\rho$ is a coherent information system and* $\mathbf{C}_{\rho\to\sigma} = {\mathbf{C}_\sigma}^{\mathbf{C}_\rho}$.

Proof. For the first statement one verifies the requirements for a c.i.s. by induction on the derivations. The second statement immediately follows from the definitions. ☐

Definition 2.4. The ideals of $\mathbf{C}_\rho$ are called *partial continuous functionals* of type ρ.

Examples. Let us consider the information systems $\mathbf{C_N}$ and $\mathbf{C_B}$. The tokens of $\mathbf{C_N}$ are the vertices in the following Hasse diagram:



Two tokens are consistent iff they have an upper bound in the diagram. A formal neighborhood U entails a token a iff there is a $b \in U$ such that b is above a (and there is a path from b to a).

For $\mathbf{C_B}$ the picture is much simpler:

$$\mathtt{tt}\ \bullet \qquad \bullet\ \mathtt{ff}$$

Definition 2.5. Each constructor C of type $\vec{\tau} \to \mu$ induces an approximable map r_C defined by:

$$r_C := \{(\vec{U}, C\vec{a}^*) \mid \vec{U} \vdash \vec{a}^*\}.$$

Here $(\vec{U}, a)$ is defined by $(\langle\rangle, a) := a$ and $(\vec{U}, U, a) := (\vec{U}, (U, a))$.

Examples. Let us write S for $|r_S|$, 0 for $\{0\}$, and $\bot$ for $\emptyset$. Then the ideals of $\mathbf{C_N}$ are

$$|\mathbf{C_N}| = \{\bot, \infty\} \cup \{S^n\bot \mid n \in \mathbb{N}\} \cup \{S^n 0 \mid n \in \mathbb{N}\},$$

where $\infty := \{S^n * \mid n \in \mathbb{N}\}$. Note that ∞ and all $S^n 0$ ($n \in \mathbb{N}$) are maximal. The ideals of $\mathbf{C_B}$ are $|\mathbf{C_B}| = \{\bot, \mathtt{tt}, \mathtt{ff}\}$, where $\mathtt{tt} = \{\mathtt{tt}\}$ and $\mathtt{ff} = \{\mathtt{ff}\}$.

Lemma 2.6. *(1) Let C be a constructor of type $\vec{\tau} \to \mu$ and $\vec{x}, \vec{y} \in |\mathbf{C}_{\vec{\tau}}|$. Then:*

$$\vec{x} \subseteq \vec{y} \leftrightarrow |r_C|(\vec{x}) \subseteq |r_C|(\vec{y}).$$

(2) Let $C_1^{\vec{\rho}\to\mu}, C_2^{\vec{\sigma}\to\mu}$ *be two distinct constructors of* μ*. Then* $|r_{C_1}|(\vec{x}_1) \cap |r_{C_2}|(\vec{x}_2) = \emptyset$ *for all* $\vec{x}_1 \in |\mathbf{C}_{\vec{\rho}}|$ *and* $\vec{x}_2 \in |\mathbf{C}_{\vec{\sigma}}|$*.*

(3) For each non-empty ideal $x \in |\mathbf{C}_\mu|$ *there exists a constructor* $C^{\vec{\tau}\to\mu}$ *and ideals* $\vec{x} \in \mathbf{C}_{\vec{\tau}}$ *with* $x = |r_C|(\vec{x})$*.*

Proof. (1) and (2) are easy to see. We now prove (3). Since $x \neq \emptyset$ there exists a constructor C and $\vec{d}^*$ with $C\vec{d}^* \in x$, say $C\colon \vec{\tau} \to \mu$ and $\vec{\tau} = \tau_1, \ldots, \tau_k$. For $1 \leq i \leq k$ define x_i by:

$$x_i := \{a \mid \exists C\vec{a}^* \in x.\, a_i^* = a\}.$$

Because x is an ideal x_i is so. We claim $x = |r_C|(\vec{x})$. One immediately verifies $x \subseteq |r_C|(\vec{x})$. For the converse let $C\vec{a}^* \in |r_C|(\vec{x})$, then there exist $\vec{U} \subseteq \vec{x}$ with $\vec{U} \vdash \vec{a}^*$. For i with $a_i^* \neq *$ we have $a_i^* \in x_i$ by the deductive closedness, so we have $C\vec{b}_i^* \in x$ with $b_{ii}^* = a_i^*$. Hence by the definition of entailment:

$$x \supseteq \{C\vec{b}_i^* \mid a_i^* \neq *\} \cup \{C\vec{d}^*\} \vdash C\vec{a}^*$$

where $C\vec{d}^*$ is only used if all a_i^* are $*$. Therefore $C\vec{a}^* \in x$ because x is an ideal. □

Lemma 2.7. *For all* $a \in C_\rho$ *we have that* $\emptyset \not\vdash_\rho a$*.*

Proof. Induction on ρ. In the case of a base type μ, this holds by definition. In the case of an arrow type $\rho \to \sigma$, we have:

$$\emptyset \vdash (U, a) \leftrightarrow \{b \mid \exists V.(V, b) \in \emptyset \land U \vdash V\} \vdash a \leftrightarrow \emptyset \vdash a.$$

Thus the claim follows from the IH. □

Lemma 2.8. *Let* $(\vec{U}, a), (\vec{V}, b) \in C_{\vec{\rho}\to\sigma}$*. Then:*

$$\{(\vec{U}, a)\} \vdash (\vec{V}, b) \leftrightarrow \vec{V} \vdash \vec{U} \land a \vdash b.$$

Proof. Use Lemma 2.7. □

3. Denotational Semantics

To each closed term M^σ as defined in Chapter 1, we associate a *denotation* – or *meaning* – as an element $[\![M]\!] \in |\mathbf{C}_\sigma|$ of the partial continuous functionals.

Definition 3.1. Let $\vec{P}$ be a constructor pattern such that $\mathsf{FV}(\vec{P}) \subseteq \vec{x}^{\vec{\sigma}}$, $\vec{x}$ distinct and $\vec{V} \in \mathrm{Con}_{\vec{\sigma}}$. Then the substitution $\vec{P}[\vec{V}/\vec{x}]$ of $\vec{V}$ for $\vec{x}$ in $\vec{P}$ is defined componentwise and by:

$$x_i[\vec{V}/\vec{x}] := V_i,$$
$$(C\vec{P})[\vec{V}/\vec{x}] := \{C\vec{b}^* \mid \text{if } P_i[\vec{V}/\vec{x}] = \emptyset \text{ then } b_i^* = *, \text{ otherwise } b_i^* \in P_i[\vec{V}/\vec{x}]\}.$$

If a constructor pattern was introduced as $\vec{P}(\vec{x})$, we will also write $\vec{P}(\vec{V})$ for $\vec{P}[\vec{V}/\vec{x}]$.

DEFINITION 3.2 (Denotation). Let M be a term with $\mathsf{FV}(M) \subseteq \vec{x}$. We inductively define the $\mathcal{P}$*–denotation* $[\![\lambda\vec{x}M]\!]^{\mathcal{P}}$ of M ($[\![\lambda\vec{x}M]\!]$ for short) by the clauses:

$$\text{DEN-VAR}\ \frac{U_i \vdash b}{(\vec{U}, b) \in [\![\lambda\vec{x}x_i]\!]} \qquad \text{DEN-APP}\ \frac{(\vec{U}, V, b) \in [\![\lambda\vec{x}M]\!] \quad (\vec{U}, V) \subseteq [\![\lambda\vec{x}N]\!]}{(\vec{U}, b) \in [\![\lambda\vec{x}.MN]\!]}$$

$$\text{DEN-C}\ \frac{\vec{V} \vdash \vec{b}^*}{(\vec{U}, \vec{V}, C\vec{b}^*) \in [\![\lambda\vec{x}C]\!]}$$

$$\text{DEN-D}\ \frac{(\vec{U}, \vec{V}, b) \in [\![\lambda\vec{x}, \vec{y}M]\!] \quad D\vec{P}(\vec{y}) \rhd M \quad \vec{W} \vdash \vec{P}(\vec{V})}{(\vec{U}, \vec{W}, b) \in [\![\lambda\vec{x}D]\!]}$$

where $(\vec{U}, V) \subseteq [\![\lambda\vec{x}N]\!]$ is short for $\forall b \in V.(\vec{U}, b) \in [\![\lambda\vec{x}N]\!]$, and recall that $(\vec{U}, b)$ is $(U_1, (U_2, \dots (U_n, b) \dots))$ for $\vec{U} = U_1, \dots, U_n$.

The *height* of a derivation $(\vec{U}, b) \in [\![\lambda\vec{x}M]\!]$ is defined as usual, i.e., as the supremum of the heights of the premises plus one. The *D-height* of a derivation $(\vec{U}, b) \in [\![\lambda\vec{x}M]\!]$ is defined similarly, namely as the supremum of all D-heights of the premises, but one is added only if the rule applied is DEN-D.

LEMMA 3.3. *The following preserve D-height:*

(1) $\vec{V} \vdash \vec{U} \wedge (\vec{U}, b) \in [\![\lambda\vec{x}M]\!] \Rightarrow (\vec{V}, b) \in [\![\lambda\vec{x}M]\!]$
(2) $\forall x \notin \mathsf{FV}(M).(\vec{U}, U, b) \in [\![\lambda\vec{z}, x.M]\!] \Leftrightarrow (\vec{U}, b) \in [\![\lambda\vec{z}M]\!]$
(3) $\forall x \notin \mathsf{FV}(M).(\vec{U}, U, b) \in [\![\lambda\vec{z}, x.Mx]\!] \Leftrightarrow (\vec{U}, U, b) \in [\![\lambda\vec{z}M]\!]$
(4) $(\vec{U}, \vec{V}, b) \in [\![\lambda\vec{x}, \vec{y}.M[\vec{P}(\vec{y})/\vec{z}]]\!] \Leftrightarrow (\vec{U}, \vec{P}(\vec{V}), b) \in [\![\lambda\vec{x}, \vec{z}.M]\!]$

PROOF. (1). $\mathrm{Ind}((\vec{U}, b) \in [\![\lambda\vec{x}M]\!])$.

(2). By induction on the derivations.

(3). $(\vec{U}, U, b) \in [\![\lambda\vec{z}, x.Mx]\!]$ comes from $(\vec{U}, U, V, b) \in [\![\lambda\vec{z}, x.M]\!]$ and $(\vec{U}, U, V) \subseteq [\![\lambda\vec{z}, x.x]\!]$ for some V. Moreover, $(\vec{U}, U, V) \subseteq [\![\lambda\vec{z}, x.x]\!]$ comes from $U \vdash V$. By (2) $(\vec{U}, V, b) \in [\![\lambda\vec{z}M]\!]$, and hence by (1) we conclude $(\vec{U}, U, b) \in [\![\lambda\vec{z}M]\!]$.

Conversely, assume $(\vec{U}, U, b) \in [\![\lambda\vec{z}M]\!]$. By (2) we obtain $(\vec{U}, U, U, b) \in [\![\lambda\vec{z}, x.M]\!]$. Moreover, $U \vdash U$ implies $(\vec{U}, U, U) \subseteq [\![\lambda\vec{z}, x.x]\!]$, so $(\vec{U}, U, b) \in [\![\lambda\vec{z}, x.Mx]\!]$ by DEN-APP.

(4). $\mathrm{Ind}(\vec{P})$. $\mathrm{Sideind}(M)$. The cases where M is C, D, or a variable among $\vec{x}$ follow immediately from (2).

Case MN. Then the following are equivalent:

$$\begin{aligned}
&(\vec{U}, \vec{V}, b) \in [\![\lambda\vec{x}, \vec{y}.(MN)[\vec{P}(\vec{y})/\vec{z}]]\!] \\
&\Leftrightarrow \exists W.(\vec{U}, \vec{V}, W, b) \in [\![\lambda\vec{x}, \vec{y}.M[\vec{P}(\vec{y})/\vec{z}]]\!] \wedge \\
&\qquad (\vec{U}, \vec{V}, W) \subseteq [\![\lambda\vec{x}, \vec{y}.N[\vec{P}(\vec{y})/\vec{z}]]\!] \\
&\overset{\text{SIH}}{\Leftrightarrow} \exists W.(\vec{U}, \vec{P}(\vec{V}), W, b) \in [\![\lambda\vec{x}, \vec{z}M]\!] \wedge \\
&\qquad (\vec{U}, \vec{P}(\vec{V}), W) \subseteq [\![\lambda\vec{x}, \vec{z}N]\!] \\
&\Leftrightarrow (\vec{U}, \vec{P}(\vec{V}), b) \in [\![\lambda\vec{x}, \vec{z}.MN]\!].
\end{aligned}$$

Case z_i. We have to show:

$$(\vec{U}, \vec{V}, b) \in [\![\lambda\vec{x}, \vec{y}.P_i(\vec{y})]\!] \Leftrightarrow P_i(\vec{V}) \vdash b.$$

For that we distinguish cases on P_i. *Case* $P_i = y_j$. Then both sides are equivalent to $V_j \vdash b$. *Case* $P_i = C\vec{Q}$. Then the following are equivalent:

$$\begin{aligned}
&(\vec{U}, \vec{V}, b) \in [\![\lambda\vec{x}, \vec{y}.C\vec{Q}(\vec{y})]\!] \\
&\Leftrightarrow (\vec{U}, \vec{V}, b) \in [\![\lambda\vec{x}, \vec{y}.(C\vec{u})[\vec{Q}(\vec{y})/\vec{u}]]\!] \\
&\overset{\text{IH}}{\Leftrightarrow} (\vec{U}, \vec{Q}(\vec{V}), b) \in [\![\lambda\vec{x}, \vec{u}.C\vec{u}]\!] \\
&\overset{(3)}{\Leftrightarrow} (\vec{U}, \vec{Q}(\vec{V}), b) \in [\![\lambda\vec{x}C]\!] \\
&\Leftrightarrow \exists\vec{b}^*(\vec{Q}(\vec{V}) \vdash \vec{b}^* \wedge C\vec{b}^* = b) \\
&\Leftrightarrow C\vec{Q}(\vec{V}) \vdash b.
\end{aligned}$$

□

DEFINITION 3.4. $U \approx V :\leftrightarrow U \vdash V \wedge V \vdash U$.

Clearly, $\approx$ is an equivalence relation on formal neighborhoods.

LEMMA 3.5. *(1) If $\vec{U} \vdash \vec{P}(\vec{V})$, there are $\vec{W}$ with $\vec{U} \approx \vec{P}(\vec{W})$ and $\vec{W} \vdash \vec{V}$.*

(2) If $\vec{P}_1[\vec{V}_1/\vec{y}_1] \approx \vec{P}_2[\vec{V}_2/\vec{y}_2]$, then $\vec{P}_1$ and $\vec{P}_2$ are unifiable and there exists $\vec{W}$ such that for $i = 1, 2$:

$$(\vec{P}_1\xi)[\vec{W}/\vec{z}] \approx \vec{P}_i[\vec{V}_i/\vec{y}_i]$$

where ξ is a most general unifier of $\vec{P}_1, \vec{P}_2$ and $\mathsf{FV}(\vec{P}_1\xi) \subseteq \vec{z}$.

PROOF. (1). Ind($\vec{P}$). *Case* x, $\langle\rangle$. Trivial.

Case $P, \vec{P}$. Use IH.

Case $C\vec{P}$. Then $C\vec{P}(\vec{V}) \neq \emptyset$, hence also $U \neq \emptyset$. Since $U \smile C\vec{P}(\vec{V})$ any $a \in U$ is of the form $a = C\vec{a}^*$ for some $\vec{a}^*$. Now define

$$U_i := \{a \mid \exists\vec{a}^*.C\vec{a}^* \in U \wedge a = a_i^*\}.$$

Let $C\vec{U}$ denote $(C\vec{x})[\vec{U}/\vec{x}]$. We first prove:

$$U \approx C\vec{U}.$$

So let $C\vec{a}^* \in C\vec{U}$. We show: $U \vdash C\vec{a}^*$. Since $U \neq \emptyset$ we can assume that not all a_i^* are $*$. For each i, either $U_i = \emptyset$ and $a_i^* = *$, or there exists $\vec{b}_i^*$ with $C\vec{b}_i^* \in U$ and $b_{ii}^* = a_i^*$. Therefore by definition:

$$U \supseteq \{C\vec{b}_i^* \mid a_i^* \neq *\} \vdash C\vec{a}^*.$$

Conversely, let $C\vec{a}^* \in U$. Define $C\vec{b}^* \in C\vec{U}$ by $b_i^* = a_i^*$ if $a_i^* \neq *$, $b_i = *$ if $U_i = \emptyset$, and otherwise (i.e., $a_i^* = *$ and $U_i \neq \emptyset$) take $b_i^* \in U_i$ arbitrarily. Clearly $\{C\vec{b}^*\} \vdash C\vec{a}^*$.

By the definition of entailment we have $\vec{U} \vdash \vec{P}(\vec{V})$, so by IH there exists $\vec{W}$ with $\vec{U} \approx \vec{P}(\vec{W})$ and $\vec{W} \vdash \vec{V}$. Thus $U \approx C\vec{U} \approx C\vec{P}(\vec{W})$.

(2). Assume $\vec{P}_1[\vec{V}_1/\vec{y}_1] \approx \vec{P}_2[\vec{V}_2/\vec{y}_2]$, then $\vec{P}_1[\vec{V}_1/\vec{y}_1] \smile \vec{P}_2[\vec{V}_2/\vec{y}_2]$, hence by the definition of consistency $\vec{P}_1$ and $\vec{P}_2$ are unifiable. We prove the rest by induction on the buildup of $\vec{P}_1, \vec{P}_2$. The cases where both are empty or a variable are trivial.

Case $P, \vec{P}$ and $Q, \vec{Q}$. By the linearity condition on the variables of constructor patterns it follows that a most general unifier of $P, \vec{P}$ and $Q, \vec{Q}$ is the union of the most general unifiers of P, Q and of $\vec{P}, \vec{Q}$. So the claim follows by IH.

Case $C\vec{P}_1$ and $C\vec{P}_2$. Then $C\vec{P}_1[\vec{V}_1/\vec{y}_1] \approx C\vec{P}_2[\vec{V}_2/\vec{y}_2]$ implies $\vec{P}_1[\vec{V}_1/\vec{y}_1] \approx \vec{P}_2[\vec{V}_2/\vec{y}_2]$. So IH suffices.

Case x and $C\vec{P}$. Say $U \approx C\vec{P}[\vec{V}/\vec{y}]$. W.l.o.g. $\xi(x) = C\vec{P}$ and therefore we can choose $\vec{W} := \vec{V}$. □

LEMMA 3.6. $[\![\lambda\vec{x}M]\!]$ *is consistent.*

PROOF. Let $(\vec{U}_i, b_i) \in [\![\lambda\vec{x}M]\!]$ $(i = 1, 2)$. We prove $(\vec{U}_1, b_1) \smile (\vec{U}_2, b_2)$ by induction on the maximum of the D-heights with a side induction on the maximum of the heights of $(\vec{U}_i, b_i) \in [\![\lambda\vec{x}M]\!]$.

Case DEN-VAR. Then $(\vec{U}_1, b_1), (\vec{U}_2, b_2) \in [\![\lambda\vec{x}x_i]\!]$ from $U_{1i} \vdash b_1$ and $U_{2i} \vdash b_2$. Now assume $\vec{U}_1 \smile \vec{U}_2$. This implies that $U_{1i} \cup U_{2i} \in \text{Con}$ and $U_{1i} \cup U_{2i} \vdash b_1, b_2$, hence $b_1 \smile b_2$.

Case DEN-APP. For $i = 1, 2$ we have:

$$\text{DEN-APP}\ \frac{(\vec{U}_i, V_i, b_i) \in [\![\lambda\vec{x}M]\!] \quad (\vec{U}_i, V_i) \subseteq [\![\lambda\vec{x}N]\!]}{(\vec{U}_i, b_i) \in [\![\lambda\vec{x}.MN]\!]}$$

Assume $\vec{U}_1 \smile \vec{U}_2$ then by SIH for the right premises we get $V_1 \smile V_2$, and hence by the SIH for the left premises $b_1 \smile b_2$.

Case DEN-C. For $i = 1, 2$ we have:

$$\text{DEN-C}\ \frac{\vec{V}_i \vdash \vec{b}_i^*}{(\vec{U}_i, \vec{V}_i, C\vec{b}_i^*) \in [\![\lambda\vec{x}C]\!]}$$

Assume $\vec{U}_1 \smile U_2$ and $\vec{V}_1 \smile \vec{V}_2$, then $\vec{V}_1 \cup \vec{V}_2 \in \text{Con}$ and $\vec{V}_1 \cup \vec{V}_2$ entails both $\vec{b}_1^*$ and $\vec{b}_2^*$. So $\vec{b}_1^* \smile \vec{b}_2^*$, i.e., $C\vec{b}_1^* \smile C\vec{b}_2^*$.

Case DEN-D. For $i = 1, 2$ we have:

$$\text{DEN-D}\ \frac{(\vec{U}_i, \vec{V}_i, b_i) \in [\![\lambda\vec{x}, \vec{y}_i M_i]\!] \quad D\vec{P}_i(\vec{y}_i) \triangleright M_i \quad \vec{W}_i \vdash \vec{P}_i[\vec{V}_i/\vec{y}_i]}{(\vec{U}_i, \vec{W}_i, b_i) \in [\![\lambda\vec{x}D]\!]}$$

Assume $\vec{U}_1 \smile \vec{U}_2$ and $\vec{W}_1 \smile \vec{W}_2$. We have to show $b_1 \smile b_2$. For $i = 1, 2$ we have that $\vec{W}_1 \cup \vec{W}_2 \vdash \vec{P}_i[\vec{V}_i/\vec{y}_i]$ so by Lemma 3.5 (1) there are $\vec{V}_1', \vec{V}_2'$ with $\vec{V}_i' \vdash \vec{V}_i$ and $\vec{W}_1 \cup \vec{W}_2 \approx \vec{P}_i[\vec{V}_i'/\vec{y}_i]$. Now using Lemma 3.5 (2) we get $\vec{W}$ with

$$(\vec{P}_1\xi)[\vec{W}/\vec{z}] \approx \vec{P}_i[\vec{V}_1'/\vec{y}_i]$$

where ξ is a most general unifier of $\vec{P}_1$ and $\vec{P}_2$. Therefore also:

$$(\vec{y}_i\xi)[\vec{W}/\vec{z}] \vdash \vec{V}_i,$$

hence by Lemma 3.3 (1) we obtain

$$(\vec{U}_i, (\vec{y}_i\xi)[\vec{W}/\vec{z}], b_i) \in [\![\lambda\vec{x}, \vec{y}_i M_i]\!]$$

with smaller D-height. With Lemma 3.3 (4) we get

$$(\vec{U}_i, \vec{W}, b_i) \in [\![\lambda\vec{x}, \vec{z}.M_i[\vec{y}_i\xi/\vec{y}_i]]\!].$$

By our restriction on the computation rules we know that $M_1\xi = M_2\xi$ and $\mathsf{FV}(M_i) \subseteq \vec{y}_i$, thus $M_1[(\vec{y}_1\xi)/\vec{y}_1] = M_1\xi = M_2\xi = M_2[(\vec{y}_2\xi)/\vec{y}_2]$. Hence we can apply the IH on $(\vec{U}_1, \vec{W}, b_1), (\vec{U}_2, \vec{W}, b_2) \in [\![\lambda\vec{x}, \vec{z}.M_1\xi]\!]$ and obtain $b_1 \smile b_2$. □

In order to prove the deductive closure of the denotation we need the following generalization of Lemma 3.5 (2).

Lemma 3.7. *Let $\vec{P}_1, \dots, \vec{P}_n$ ($n \geq 1$) be constructor patterns and assume for all $1 \leq i, j \leq n$ that $\vec{P}_i[\vec{V}_i/\vec{y}_i] \approx \vec{P}_j[\vec{V}_j/\vec{y}_j]$ and either $\vec{P}_i = \vec{P}_j$ or $\mathsf{FV}(\vec{P}_i) \cap \mathsf{FV}(\vec{P}_j) = \emptyset$. Then there exist $\vec{W}$ and a substitution ξ such that for all $1 \leq i \leq n$:*

$$\xi \text{ is admissible for } \vec{P}_i,\ \vec{P}_1\xi = \vec{P}_i\xi, \text{ and } (\vec{P}_1\xi)[\vec{W}/\vec{z}] \approx \vec{P}_i[\vec{V}_i/\vec{y}_i].$$

Proof. Ind(n). *Case* $n = 1$. Trivial. *Case* $n \to n+1$. By Lemma 3.5 (2) we may assume that $n \geq 2$. So by IH there exist $\vec{W}$ and ξ such that for all $1 \leq i \leq n$: ξ is admissible for $\vec{P}_i$, $\vec{P}_1\xi = \vec{P}_i\xi$, and $(\vec{P}_1\xi)[\vec{W}/\vec{z}] \approx P_i[\vec{V}_i/\vec{y}_i]$. If $\vec{P}_{n+1} = \vec{P}_i$ for some $1 \leq i \leq n$, we are already done. So assume $\vec{P}_{n+1}$ is different from $\vec{P}_i$ ($1 \leq i \leq n$). Then by assumption $\vec{P}_{n+1}$ has different variables than all $\vec{P}_i$ ($1 \leq i \leq n$). Therefore we can assume w.l.o.g. that $\mathsf{FV}(\vec{P}_1\xi) \cap \mathsf{FV}(\vec{P}_{n+1}) = \emptyset$ and $\vec{P}_{n+1}\xi = \vec{P}_{n+1}$ (otherwise rename those variables in the image of ξ and restrict ξ to variables not free in $\vec{P}_{n+1}$). Since $n \geq 2$ we have

$$(\vec{P}_1\xi)[\vec{W}/\vec{z}] \approx \vec{P}_2[\vec{V}_2/\vec{y}_2] \approx \vec{P}_{n+1}[\vec{V}_{n+1}/\vec{y}_{n+1}]$$

by assumption. By Lemma 3.5 (2) there are $\vec{U}$ such that for a most general unifier η of $\vec{P}_1\xi$ and $\vec{P}_{n+1}$ we have

$$((\vec{P}_1\xi)\eta)[\vec{U}/\vec{x}] \approx (\vec{P}_1\xi)[\vec{W}/\vec{z}] \quad \text{and} \quad ((\vec{P}_1\xi)\eta)[\vec{U}/\vec{x}] \approx \vec{P}_{n+1}[\vec{V}_{n+1}/\vec{y}_{n+1}].$$

Thus for $1 \leq i \leq n$:

$$((\vec{P}_1\xi)\eta)[\vec{U}/\vec{x}] \approx (\vec{P}_1\xi)[\vec{W}/\vec{z}] \approx \vec{P}_i[\vec{V}_i/\vec{y}_i] \quad \text{and} \quad (\vec{P}_1\xi)\eta = (\vec{P}_i\xi)\eta.$$

Moreover, by our assumption on ξ:

$$(\vec{P}_{n+1}\xi)\eta = \vec{P}_{n+1}\eta = (\vec{P}_1\xi)\eta.$$

The substitution $\eta \circ \xi$ is admissible for all $\vec{P}_i$ ($1 \leq i \leq n+1$) because $(\vec{P}_1\xi)\eta$ is a constructor pattern. So take $\vec{U}$ and $\eta \circ \xi$. □

Lemma 3.8. *$[\![\lambda\vec{x}M]\!]$ is deductively closed, i.e.,*

$$W \in \mathrm{Con} \wedge W \subseteq [\![\lambda\vec{x}M]\!] \wedge W \vdash (\vec{V}, c) \to (\vec{V}, c) \in [\![\lambda\vec{x}M]\!].$$

Proof. Induction on the maximum of the D-heights with a side induction on the maximum of the heights of $W \subseteq [\![\lambda\vec{x}M]\!]$. We make a case distinction on the last rule of the derivations (which is unique since the definition is syntax directed, i.e., depends only on M).

Case den-var. For all $(\vec{U}, b) \in W$ we have

$$\textsc{den-var}\ \frac{U_i \vdash b}{(\vec{U}, b) \in [\![\lambda\vec{x}x_i]\!]}.$$

We must show $V_i \vdash c$. By assumption we have $W \vdash (\vec{V}, c)$, i.e., $W(\vec{V}) \vdash c$. So it suffices to show

$$V_i \vdash W(\vec{V}).$$

Let $b \in W(\vec{V})$, then there exist $\vec{U}$ with $(\vec{U}, b) \in W$ and $\vec{V} \vdash \vec{U}$. But $(\vec{U}, b) \in W$ implies $U_i \vdash b$, so $V_i \vdash U_i \vdash b$.

Case DEN-APP. For $W = \{(\vec{U}_1, b_1), \dots, (\vec{U}_n, b_n)\}$ and each $(\vec{U}_i, b_i) \in W$ there exists U_i such that

$$\text{DEN-APP}\ \frac{(\vec{U}_i, U_i, b_i) \in [\![\lambda\vec{x}M]\!] \quad (\vec{U}_i, U_i) \subseteq [\![\lambda\vec{x}N]\!]}{(\vec{U}_i, b_i) \in [\![\lambda\vec{x}.MN]\!]}.$$

Define $U := \bigcup\{U_i \mid \vec{V} \vdash \vec{U}_i\}$. First, we show that U is consistent. For $a, b \in U$ there exist i, j such that $a \in U_i$, $b \in U_j$, and $\vec{V} \vdash \vec{U}_i, \vec{U}_j$. Then $\vec{U}_i \smile \vec{U}_j$, so because $[\![\lambda\vec{x}N]\!]$ is consistent by Lemma 3.6 and $(\vec{U}_i, a), (\vec{U}_j, b) \in [\![\lambda\vec{x}N]\!]$ we obtain $a \smile b$. So $U \in \text{Con}$.

Next we prove

$$(\vec{V}, U) \subseteq [\![\lambda\vec{x}N]\!].$$

Let $a \in U$; then $a \in U_i$ for some i with $\vec{V} \vdash \vec{U}_i$. Let $W' := \{(\vec{U}_i, b) \mid b \in U_i\}$, then by our premise $W' \subseteq [\![\lambda\vec{x}N]\!]$ and therefore also $W' \in \text{Con}$. By SIH it suffices to show that $W' \vdash (\vec{V}, a)$, i.e., $W'(\vec{V}) \vdash a$. But

$$W'(\vec{V}) = \{b \mid b \in U_i \wedge \vec{V} \vdash \vec{U}_i\} = U_i \vdash a.$$

Finally we show

$$(\vec{V}, U, c) \in [\![\lambda\vec{x}M]\!].$$

Define $W'' := \{(\vec{U}_i, U_i, b_i) \mid 1 \leq i \leq n\}$. By assumption $W'' \subseteq [\![\lambda\vec{x}M]\!]$, hence also $W'' \in \text{Con}$. By SIH it is enough to prove that $W'' \vdash (\vec{V}, U, c)$, i.e., $W''(\vec{V}, U) \vdash c$. Now:

$$\begin{aligned} W''(\vec{V}, U) &= \{b_i \mid \vec{V} \vdash \vec{U}_i \wedge U \vdash U_i\} \\ &= \{b_i \mid \vec{V} \vdash \vec{U}_i\} && \vec{V} \vdash \vec{U}_i \text{ implies } U \supseteq U_i \vdash U_i \\ &= W(\vec{V}). \end{aligned}$$

And $W(\vec{V}) \vdash c$ since $W \vdash (\vec{V}, c)$.

Altogether we use DEN-APP and get $(\vec{V}, c) \in [\![\lambda\vec{x}.MN]\!]$ as required.

Case DEN-C. For $(\vec{U}, \vec{U}', C\vec{b}^*) \in W$ we have

$$\text{DEN-C}\ \frac{\vec{U}' \vdash \vec{b}^*}{(\vec{U}, \vec{U}', C\vec{b}^*) \in [\![\lambda\vec{x}C]\!]}.$$

We have to show that $(\vec{V}, \vec{V}', c) \in [\![\lambda\vec{x}C]\!]$. By assumption $W(\vec{V}, \vec{V}') \vdash c$, i.e.,

$$\{C\vec{b}^* \mid \exists \vec{U}, \vec{U}'.(\vec{U}, \vec{U}', C\vec{b}^*) \in W \wedge \vec{V} \vdash \vec{U} \wedge \vec{V}' \vdash \vec{U}'\} \vdash c.$$

So by the definition of entailment there exists $\vec{c}^*$ with $c = C\vec{c}^*$ and

$$W_i := \{b \mid \exists \vec{U}, \vec{U}', \vec{b}^*. b_i^* = b \wedge (\vec{U}, \vec{U}', C\vec{b}^*) \in W \wedge \vec{V} \vdash \vec{U} \wedge \vec{V}' \vdash \vec{U}'\} \vdash c_i^*.$$

We need $\vec{V}' \vdash \vec{c}^*$. Given i, it suffices to prove $V_i' \vdash W_i$. Let $b \in W_i$, then there exist $\vec{U}, \vec{U}'$, and $\vec{b}^*$ such that $b_i^* = b$, $(\vec{U}, \vec{U}', C\vec{b}^*) \in W$ and $\vec{V}' \vdash \vec{U}'$. Therefore, since $\vec{U}' \vdash \vec{b}^*$:

$$V_i' \vdash U_i' \vdash b_i^* = b.$$

Case DEN-D. Let $W = \{(\vec{U}_1, \vec{U}_1'', b_1), \dots, (\vec{U}_n, \vec{U}_n'', b_n)\}$. Then for each i there exist $\vec{U}_i'$ such that

$$\text{DEN-D}\ \frac{(\vec{U}_i, \vec{U}_i', b_i) \in [\![\lambda\vec{x}, \vec{y}_i M_i]\!] \quad D\vec{P}_i(\vec{y}_i) \rhd M_i \quad \vec{U}_i'' \vdash \vec{P}_i[\vec{U}_i'/\vec{y}_i]}{(\vec{U}_i, \vec{U}_i'', b_i) \in [\![\lambda\vec{x}D]\!]}.$$

Assume $W \vdash (\vec{V}, \vec{V}'', c)$. We have to prove $(\vec{V}, \vec{V}'', c) \in [\![\lambda\vec{x}D]\!]$. Define

$$I := \{i \mid 1 \le i \le n \wedge \vec{V} \vdash \vec{U}_i \wedge \vec{V}'' \vdash \vec{U}_i''\}.$$

Then $\{b_i \mid i \in I\} = W(\vec{V}, \vec{V}'') \vdash c$, so by Lemma 2.7 $I \neq \emptyset$, w.l.o.g. $1 \in I$. For $i \in I$ we have $\vec{V}'' \vdash \vec{U}_i'' \vdash \vec{P}_i[\vec{U}_i'/\vec{y}_i]$, and therefore by Lemma 3.5 (1) there are $\vec{V}_i'$ with $\vec{V}_i' \vdash U_i'$ and $\vec{V}'' \approx \vec{P}_i[\vec{V}_i'/\vec{y}_i]$. In particular, for all $i, j \in I$:

$$\vec{P}_i[\vec{V}_i'/\vec{y}_i] \approx \vec{V}'' \approx \vec{P}_j[\vec{V}_j'/\vec{y}_j].$$

Hence $\vec{P}_i$ and $\vec{P}_j$ are unifiable and thus, by our restriction on the computation rules, are either equal or have distinct variables. So we have verified that we can apply Lemma 3.7 and get a substitution ξ and $\vec{W}$ such that for all $i \in I$:

$$P_1\xi = P_i\xi \quad \text{and} \quad (\vec{P}_1\xi)[\vec{W}/\vec{z}] \approx \vec{P}_i[\vec{V}_i'/\vec{y}_i].$$

Let $i \in I$. Then ξ factors through the most general unifier of $\vec{P}_1$ and $\vec{P}_i$, so we get $M_1\xi = M_i\xi$. Furthermore, $(\vec{y}_i\xi)[\vec{W}/\vec{z}] \vdash \vec{V}_i' \vdash \vec{U}_i'$. Applying Lemma 3.3 (1) we get

$$(\vec{V}, (\vec{y}_i\xi)[\vec{W}/\vec{z}], b_i) \in [\![\lambda\vec{x}, \vec{y}_i M_i]\!].$$

Therefore by Lemma 3.3 (4) we obtain (with a smaller D-height)

$$(\vec{V}, \vec{W}, b_i) \in [\![\lambda\vec{x}, \vec{z}.M_i[\vec{y}_i\xi/\vec{y}_i]]\!].$$

But $M_i[\vec{y}_i\xi/\vec{y}_i] = M_i\xi = M_1\xi = M_1[\vec{y}_1\xi/\vec{y}_1]$ and thus we have

$$\forall i \in I.(\vec{V}, \vec{W}, b_i) \in [\![\lambda\vec{x}, \vec{z}.M_1[\vec{y}_1\xi/\vec{y}_1]]\!].$$

Now $X := \{(\vec{V}, \vec{W}, b_i) \mid i \in I\} \subseteq [\![\lambda\vec{x}, \vec{z}.M_1[\vec{y}_1\xi/\vec{y}_1]]\!]$ is a formal neighborhood by Lemma 3.6 and

$$X(\vec{V}, \vec{W}) = \{b_i \mid i \in I\} = W(\vec{V}, \vec{V}'') \vdash c.$$

So by IH we get that $(\vec{V}, \vec{W}, c) \in [\![\lambda\vec{x}, \vec{z}.M_1[\vec{y}_1\xi/\vec{y}_1]]\!]$. By Lemma 3.3 (4) we obtain $(\vec{V}, (y_1\xi)[\vec{W}/\vec{z}], c) \in [\![\lambda\vec{x}, \vec{y}_1 M_1]\!]$. Together with $\vec{V}'' \approx (\vec{P}_1\xi)[\vec{W}/\vec{z}] = \vec{P}_1[(\vec{y}_1\xi)[\vec{W}/\vec{z}]/\vec{y}_1]$ and DEN-D we can conclude $(\vec{V}, \vec{V}'', c) \in [\![\lambda\vec{x}D]\!]$. □

COROLLARY 3.9. *$[\![\lambda\vec{x}M]\!]$ is an ideal.*

4. Preservation of Values

In this section we show that our reduction relation $\longrightarrow$ is correct w.r.t. our denotational semantics (see Section 3), in the sense that, if a term reduces to another one, their denotations are equal. It is convenient to extend the denotation to open terms (i.e., terms with free variables) via an assignment of the free variables.

Definition 4.1. (1) Let $\theta\colon \mathsf{Var} \to \bigcup_{\tau\in\mathsf{Ty}} |\mathbf{C}_\tau|$. Then θ is called an *environment of ideals* if $\theta(x^\tau) \in |\mathbf{C}_\tau|$ for each $x^\tau \in \mathsf{Var}_\tau$ and the support $\mathsf{supp}(\theta) := \{x \mid \theta(x) \neq \emptyset\}$ is finite. We usually write θ for an environment of ideals.

(2) Let $\vec{x}^{\vec{\rho}}$ be distinct variables, $\vec{v} \in |\mathbf{C}_{\vec{\rho}}|$ be ideals, and θ an environment of ideals. The *update* $\theta[\vec{x} \mapsto \vec{v}]$ of θ is defined as

$$(\theta[\vec{x} \mapsto \vec{v}])(x) := \begin{cases} v_i & \text{if } x = x_i, \\ \theta(x) & \text{otherwise.} \end{cases}$$

(3) $\theta \subseteq \theta' :\leftrightarrow \forall x \in \mathsf{Var}(\theta(x) \subseteq \theta'(x))$.

(4) For $\mathsf{FV}(M) \subseteq \vec{x}$ we define

$$[\![M]\!]_\theta := \{b \mid \exists \vec{U} \subseteq \theta(\vec{x})\, (\vec{U}, b) \in [\![\lambda\vec{x}M]\!]\},$$

where $\theta(\vec{x}) = \theta(x_1), \ldots, \theta(x_n)$ for $\vec{x} = x_1, \ldots, x_n$.

Note that, by Lemma 3.3 (1), $[\![M]\!]_\theta$ does not depend on the choice of $\vec{x}$, and hence is well defined. Moreover, we get the following:

Lemma 4.2 (Coincidence).

$$\forall x \in \mathsf{FV}(M)(\theta(x) = \theta'(x)) \to [\![M]\!]_\theta = [\![M]\!]_{\theta'}.$$

Lemma 4.3 (Monotonicity).

$$\theta' \supseteq \theta \wedge U \subseteq [\![M]\!]_\theta \wedge U \vdash a \to a \in [\![M]\!]_{\theta'}.$$

Proof. Let $\mathsf{FV}(M) \subseteq \vec{x}$. Then we have

$$\forall b \in U \exists \vec{U} \subseteq \theta(\vec{x}).(\vec{U}, b) \in [\![\lambda\vec{x}M]\!].$$

Now since U is finite, $\theta(\vec{x})$ and $[\![\lambda\vec{x}M]\!]$ are ideals, we get $\vec{W} \subseteq \theta(\vec{x})$ (taking for $\vec{W}$ the union of all $\vec{U}$'s) so that:

$$\forall b \in U.(\vec{W}, b) \in [\![\lambda\vec{x}M]\!].$$

But $\{(\vec{W}, b) \mid b \in U\} \vdash (\vec{W}, a)$ since $U \vdash a$. Therefore $(\vec{W}, a) \in [\![\lambda\vec{x}M]\!]$ by the deductive closure of $[\![\lambda\vec{x}M]\!]$. So $a \in [\![M]\!]_\theta \subseteq [\![M]\!]_{\theta'}$. □

Corollary 4.4. *$[\![M]\!]_\theta$ is an ideal.*

Proof. We only need to check consistency. Let $\mathsf{FV}(M) \subseteq \vec{x}$ and $a, b \in [\![M]\!]_\theta$. By definition there are $\vec{U}, \vec{V} \subseteq \theta(\vec{x})$ with $(\vec{U}, a), (\vec{V}, b) \in [\![\lambda\vec{x}M]\!]$. Because $\theta(\vec{x})$ are ideals we obtain $\vec{U} \smile \vec{V}$. So $a \smile b$ since $[\![\lambda\vec{x}M]\!]$ is an ideal. □

Lemma 4.5. *(1) $[\![x]\!]_\theta = \theta(x)$.*

(2) $[\![\lambda xM]\!]_\theta = \{(V, b) \mid b \in [\![M]\!]_{\theta[x \mapsto \overline{V}]}\}$.

(3) $[\![MN]\!]_\theta = [\![M]\!]_\theta [\![N]\!]_\theta$.

Proof. (1). Easy.

(2). Let $\mathsf{FV}(\lambda x M) \subseteq \vec{x}$, w.l.o.g. x is distinct from $\vec{x}$. Then:

$$\begin{aligned}
[\![\lambda x M]\!] &= \{(V,b) \mid \exists \vec{U} \subseteq \theta(\vec{x}).(\vec{U},V,b) \in [\![\lambda \vec{x}, x M]\!]\} \\
&\overset{(*)}{=} \{(V,b) \mid \exists \vec{U} \subseteq \theta(\vec{x}) \exists W.V \vdash W \wedge (\vec{U},W,b) \in [\![\lambda \vec{x}, x M]\!]\} \\
&= \{(V,b) \mid \exists \vec{U}, W \subseteq (\theta[x \mapsto \overline{V}])(\vec{x},x).(\vec{U},W,b) \in [\![\lambda \vec{x}, x M]\!]\} \\
&= \{(V,b) \mid b \in [\![M]\!]_{\theta[x \mapsto \overline{V}]}\},
\end{aligned}$$

where at $(*)$ we have used that $[\![\lambda \vec{x}, x M]\!]$ is deductively closed.

(3). Let $\mathsf{FV}(MN) \subseteq \vec{x}$. Then:

$$\begin{aligned}
b \in [\![MN]\!]_\theta &\leftrightarrow \exists \vec{U} \subseteq \theta(\vec{x}).(\vec{U},b) \in [\![\lambda \vec{x}.MN]\!] \\
&\leftrightarrow \exists \vec{U} \subseteq \theta(\vec{x}) \exists U.(\vec{U},U,b) \in [\![\lambda \vec{x} M]\!] \wedge (\vec{U},U) \subseteq [\![\lambda \vec{x} N]\!] \\
&\leftrightarrow \exists U \exists \vec{U} \subseteq \theta(\vec{x}).(\vec{U},U,b) \in [\![\lambda \vec{x} M]\!] \wedge (\vec{U},U) \subseteq [\![\lambda \vec{x} N]\!] \\
&\leftrightarrow \exists U.(U,b) \in [\![M]\!]_\theta \wedge U \subseteq [\![N]\!]_\theta \\
&\leftrightarrow b \in [\![M]\!]_\theta [\![N]\!]_\theta,
\end{aligned}$$

where in the upwards direction of the second to last equivalence, we have used that the $\theta(x)$'s are ideals. □

Lemma 4.6. $[\![\lambda x M]\!]_\theta(v) = [\![M]\!]_{\theta[x \mapsto v]}$.

Proof. Let $\mathsf{FV}(M) \subseteq \vec{x}$ and assume w.l.o.g. x is distinct from $\vec{x}$. Then:

$$\begin{aligned}
b \in [\![\lambda x M]\!]_\theta(v) &\leftrightarrow \exists V \subseteq v.(V,b) \in [\![\lambda x M]\!]_\theta \\
&\leftrightarrow \exists V \subseteq v \exists \vec{U} \subseteq \theta(\vec{x}).(\vec{U},V,b) \in [\![\lambda \vec{x}, x M]\!] \\
&\leftrightarrow \exists \vec{U}, V \subseteq (\theta[x \mapsto v])(\vec{x},x).(\vec{U},V,b) \in [\![\lambda \vec{x}, x M]\!] \\
&\leftrightarrow b \in [\![\lambda x M]\!]_{\theta[x \mapsto v]}.
\end{aligned}$$ □

Lemma 4.7 (Substitution). $[\![M]\!]_{\theta[\vec{z} \mapsto [\![\vec{N}]\!]_\theta]} = [\![M[\vec{N}/\vec{z}]]\!]_\theta$.

Proof. Ind(M). The cases where M is a constant are trivial. The application and variable cases follow easily from Lemma 4.5 using IH in the former. For the case $\lambda x M$ assume $\mathsf{FV}(\lambda x M) \subseteq \vec{x}$ and w.l.o.g. $x \notin \mathsf{FV}(\vec{N}, \vec{z})$. By the Coincidence Lemma we get

$$[\![\vec{N}]\!]_\theta = [\![\vec{N}]\!]_{\theta[x \mapsto \overline{U}]}.$$

Using Lemma 4.5 (2) we conclude:

$$\begin{aligned}
(U,b) \in [\![\lambda x M]\!]_{\theta[\vec{z} \mapsto [\![\vec{N}]\!]_\theta]} &\leftrightarrow b \in [\![M]\!]_{\theta[\vec{z} \mapsto [\![\vec{N}]\!]_\theta][x \mapsto \overline{U}]} \\
&\leftrightarrow b \in [\![M]\!]_{\theta[x \mapsto \overline{U}][\vec{z} \mapsto [\![\vec{N}]\!]_{\theta[x \mapsto \overline{U}]}]} \\
&\overset{\mathrm{IH}}{\leftrightarrow} b \in [\![M[\vec{N}/\vec{z}]]\!]_{\theta[x \mapsto \overline{U}]} \\
&\leftrightarrow (U,b) \in [\![\lambda x.M[\vec{N}/\vec{z}]]\!]_\theta.
\end{aligned}$$ □

Corollary 4.8 (Preservation under β). $[\![(\lambda x M)N]\!]_\theta = [\![M[N/x]]\!]_\theta$.

Proof. Using the preceding lemmas we calculate:

$$[\![(\lambda x M)N]\!]_\theta = [\![\lambda x M]\!]_\theta [\![N]\!]_\theta = [\![M]\!]_{\theta[x \mapsto [\![N]\!]_\theta]} = [\![M[N/x]]\!]_\theta.$$ □

LEMMA 4.9 (Preservation under η). *If $x \notin \mathsf{FV}(M)$, then*

$$[\![\lambda x.Mx]\!]_\theta = [\![M]\!]_\theta.$$

PROOF. Use Lemma 3.3 (3). □

LEMMA 4.10. *Let $D\vec{P}(\vec{y}) \triangleright M$ be a computation rule. Then:*

$$[\![\lambda\vec{y}.D\vec{P}(\vec{y})]\!] = [\![\lambda\vec{y}M]\!].$$

PROOF. First observe that

$$(\vec{V}, b) \in [\![\lambda\vec{y}.D\vec{P}(\vec{y})]\!] \leftrightarrow (\vec{P}(\vec{V}), b) \in [\![\lambda\vec{z}.D\vec{z}]\!] = [\![D]\!]$$

by Lemma 3.3 (4) and (3). Now if $(\vec{P}(\vec{V}), b) \in [\![D]\!]$, there exist $\vec{U}$ such that $(\vec{U}, b) \in [\![\lambda\vec{y}M]\!]$ and $\vec{P}(\vec{V}) \vdash \vec{P}(\vec{U})$, hence also $\vec{V} \vdash \vec{U}$. By the deductive closure of the denotation we obtain $(\vec{V}, b) \in [\![\lambda\vec{y}M]\!]$.

Conversely, $(\vec{V}, b) \in [\![\lambda\vec{y}M]\!]$ yields $(\vec{P}(\vec{V}), b) \in [\![D]\!]$ by DEN-D. □

COROLLARY 4.11. *If $M \longrightarrow N$, then $[\![M]\!]_\theta = [\![N]\!]_\theta$.*

PROOF. By induction on $M \longrightarrow N$ using the preceding lemmas. □

5. Operational Semantics and Computational Adequacy

In this section we introduce a deterministic reduction relation $\longrightarrow_{\mathsf{d}}$ – our *operational semantics* –, which is computationally adequate w.r.t. the denotational semantics. This asserts that denotational semantics and operational semantics coincide, in the sense that a closed term M is denotationally equal to a numeral if and only if it reduces with $\longrightarrow_{\mathsf{d}}^*$ to that numeral.

In the following B ranges over constructors and defined constants. Let $\mathsf{rdx}_{\mathcal{P}}$ be the set of all $\mathcal{P}$-redexes.

DEFINITION 5.1 ($\mathsf{nf}_{\mathsf{d}}^{\mathcal{P}}$). We inductively define $M \in \mathsf{nf}_{\mathsf{d}}^{\mathcal{P}}$, where we write nf for $\mathsf{nf}_{\mathsf{d}}^{\mathcal{P}}$:

$$\frac{}{x\vec{M} \in \mathsf{nf}} \qquad \frac{}{\lambda x M \in \mathsf{nf}} \qquad \frac{|\vec{M}| < |\mathsf{ar}(B)|}{B\vec{M} \in \mathsf{nf}}$$

$$\frac{\vec{M} \in \mathsf{nf} \quad |\vec{M}| = |\mathsf{ar}(B)| \quad B\vec{M} \notin \mathsf{rdx}_{\mathcal{P}}}{B\vec{M}\vec{N} \in \mathsf{nf}}$$

DEFINITION 5.2 ($\longrightarrow_{\mathsf{d}}^{\mathcal{P}}$). The relation $\longrightarrow_{\mathsf{d}}^{\mathcal{P}}$ is inductively defined by:

$$\text{PRED-BETA}\ \frac{}{(\lambda x M)N \longrightarrow_{\mathsf{d}} M[N/x]} \qquad \text{PRED-D}\ \frac{D\vec{P}(\vec{y}) \triangleright M}{D\vec{P}(\vec{N}) \longrightarrow_{\mathsf{d}} M[\vec{N}/\vec{y}]}$$

$$\text{PRED-CONG-L}\ \frac{M \longrightarrow_{\mathsf{d}} M'}{MN \longrightarrow_{\mathsf{d}} M'N}$$

$$\text{PRED-CONST-PAR}\ \frac{\vec{M} \longrightarrow_{\mathsf{d}!} \vec{N} \quad |\vec{M}| = |\mathsf{ar}(B)| \quad \exists i(M_i \notin \mathsf{nf}) \quad B\vec{M} \notin \mathsf{rdx}_{\mathcal{P}}}{B\vec{M} \longrightarrow_{\mathsf{d}} B\vec{N}}$$

where we have written $\longrightarrow_{\mathsf{d}}$ for $\longrightarrow_{\mathsf{d}}^{\mathcal{P}}$, and $M \longrightarrow_{\mathsf{d}!} N$ is defined as either $M \in \mathsf{nf}$ and $M = N$, or $M \longrightarrow_{\mathsf{d}} N$. We write $M \not\longrightarrow_{\mathsf{d}}$ for $\forall N \neg(M \longrightarrow_{\mathsf{d}} N)$, i.e., M is normal w.r.t. $\longrightarrow_{\mathsf{d}}$.

LEMMA 5.3. *(1) $M \not\longrightarrow_{\mathsf{d}} \Leftrightarrow M \in \mathsf{nf}$.*

(2) The relation $\longrightarrow_{\mathsf{d}}$ *is deterministic, i.e.,*

$$M \longrightarrow_{\mathsf{d}} N_1, N_2 \Rightarrow N_1 = N_2.$$

(3) $\longrightarrow_{\mathsf{d}} \subseteq \longrightarrow^*$.

Proof. The first statement is proved by induction on M.

For the second statement observe that $M \longrightarrow_{\mathsf{d}} N_i$ must be derived by the same rule for $i = 1, 2$ (by case distinction on M using (1)). Now the claim follows by induction on $M \longrightarrow_{\mathsf{d}} N_1, N_2$, where in the case PRED-D we need the restriction for overlapping computation rules.

The last statement is easily proved by induction on $M \longrightarrow_{\mathsf{d}} N$. □

Computational Adequacy. One direction of adequacy is the correctness of $\longrightarrow_{\mathsf{d}}$ (i.e., $\longrightarrow_{\mathsf{d}}$ preserves values), which immediately follows from the fact that $\longrightarrow_{\mathsf{d}}$ is contained in $\longrightarrow^*$ and the latter preserves values. We now prove the other direction by means of an "operational interpretation" (cf. [**28**]). The Adequacy Theorem was first proved by Plotkin for PCF in [**33**] using computability predicates; our proof is based on [**36**] but adapted to our setting.

Definition 5.4. We inductively define when a closed term M of type τ belongs to the *operational interpretation* of a token a of type τ, $M \in [a]$, where for a formal neighborhood U of type τ, $M \in [U]$ is defined as $\forall a \in U(M \in [a])$, and $M \in [*]$ is true by definition. The rules are:

$$\text{OINT-BASE}\ \frac{\vec{M} \in [\vec{b}^*] \quad M \longrightarrow_{\mathsf{d}}^* C\vec{M}}{M \in [C\vec{b}^*]}$$

$$\text{OINT-ARR}\ \frac{M \longrightarrow_{\mathsf{d}}^* M' \not\longrightarrow_{\mathsf{d}} \quad \forall N \in [U](MN \in [b])}{M \in [(U,b)]}$$

Lemma 5.5 (Monotonicity and Expansion). *Let M be closed. Then:*

$$M \longrightarrow_{\mathsf{d}}^* N \wedge N \in [V] \wedge V \vdash U \Rightarrow M \in [U].$$

Proof. Induction on the type of M with a side induction on $N \in [V]$. Let $b \in U$; we have to prove $M \in [b]$.

Case μ. Then $V = \{C\vec{a}_1^*, \dots, C\vec{a}_n^*\}$ $(n \geq 1)$ and $b = C\vec{b}^*$ with

$$\{a_{1j}^*, \dots, a_{nj}^*\} \vdash b_j^*.$$

And for each $1 \leq i \leq n$ there are $\vec{N}_i$ with

$$\text{OINT-BASE}\ \frac{\vec{N}_i \in [\vec{a}_i^*] \quad N \longrightarrow_{\mathsf{d}}^* C\vec{N}_i}{N \in [C\vec{a}_i^*]}.$$

Since $\longrightarrow_{\mathsf{d}}$ is deterministic there exists i_0 such that for all $1 \leq i \leq n$, $C\vec{N}_{i_0} \longrightarrow_{\mathsf{d}}^* C\vec{N}_i$, and hence $\vec{N}_{i_0} \longrightarrow_{\mathsf{d}}^* \vec{N}_i \in [\vec{a}_i^*]$. By SIH we get $\vec{N}_{i_0} \in [\vec{b}^*]$. Moreover, $M \longrightarrow_{\mathsf{d}}^* N \longrightarrow_{\mathsf{d}}^* C\vec{N}_{i_0}$ and thus $M \in [C\vec{b}^*]$ by OINT-BASE.

Case $\rho \to \sigma$. Then $b = (U', b')$ with $V(U') \vdash b'$. Moreover, there exists N' such that for all $(V', a') \in V$:

$$\text{OINT-ARR}\ \frac{N \longrightarrow_{\mathsf{d}}^* N' \not\longrightarrow_{\mathsf{d}} \quad \forall K \in [V'](NK \in [a'])}{N \in [(V', a')]}.$$

It is enough to prove that

$$\forall K \in [U'](MK \in [b']).$$

Let $K \in [U']$. By IH and $V(U') \vdash b'$ it suffices to show $MK \in [V(U')]$. So let $a' \in V(U')$. Then there is a V' with $(V', a') \in V$ and $U' \vdash V'$. From the IH we get $K \in [V']$, and hence $NK \in [a']$. Moreover, by PRED-CONG-L, $MK \longrightarrow_{\mathsf{d}}^{*} NK$, so the IH yields $MK \in [a']$ as required. □

COROLLARY 5.6 (Closure under reduction).

$$M \in [a] \wedge M \longrightarrow_{\mathsf{d}} N \Rightarrow N \in [a].$$

PROOF. Induction on $M \in [a]$.

Case OINT-BASE.

$$\frac{\vec{M} \in [\vec{b}^*] \quad M \longrightarrow_{\mathsf{d}}^{*} C\vec{M}}{M \in [C\vec{b}^*]}$$

Then also $C\vec{M} \in [C\vec{b}^*]$ and because $\longrightarrow_{\mathsf{d}}$ is deterministic either $N \longrightarrow_{\mathsf{d}}^{*} C\vec{M}$ or $C\vec{M} \longrightarrow_{\mathsf{d}} C\vec{N} = N$ with $\vec{M} \longrightarrow_{\mathsf{d!}} \vec{N}$. In the first case we are done by the Expansion Lemma, so assume the latter. By IH we have $\vec{N} \in [\vec{b}^*]$, hence $N = C\vec{N} \in [C\vec{b}^*]$.

Case OINT-ARR.

$$\frac{M \longrightarrow_{\mathsf{d}}^{*} M' \not\longrightarrow_{\mathsf{d}} \quad \forall K \in [U](MK \in [b])}{M \in [(U, b)]}$$

For $K \in [U]$ we have $MK \in [b]$ and $MK \longrightarrow_{\mathsf{d}}^{*} M'K$. Hence by IH $M'K \in [b]$ and therefore $M' \in [(U, b)]$. Since $M' \not\longrightarrow_{\mathsf{d}}$ we conclude $M \longrightarrow_{\mathsf{d}} N \longrightarrow_{\mathsf{d}}^{*} M'$, so $N \in [(U, b)]$ by the Expansion Lemma. □

LEMMA 5.7. $\lambda\vec{x}M \in [(\vec{U}, b)] \Leftrightarrow \forall \vec{K} \in [\vec{U}](M[\vec{K}/\vec{x}] \in [b])$.

PROOF. The corollary and the lemma before imply that

$$(\lambda x M)K \in [b] \Leftrightarrow M[K/x] \in [b].$$

Using this the claim follows by induction on the length of $\vec{x}$. □

LEMMA 5.8. *(1)* $\forall \vec{L} \in [\vec{V}](C\vec{L} \in [C\vec{b}^*]) \Rightarrow C \in [(\vec{V}, C\vec{b}^*)]$

(2) $\vec{L} \longrightarrow_{\mathsf{d}}^{\leq m} \vec{P}(\vec{N}) \Rightarrow \forall n \geq m \exists \vec{N}'.\vec{N} \longrightarrow_{\mathsf{d}}^{*} \vec{N}' \wedge \vec{L} \longrightarrow_{\mathsf{d!}}^{n} \vec{P}(\vec{N}')$

(3) $\vec{L} \in [\vec{P}(\vec{b}^*)] \Rightarrow \exists \vec{N} \in [\vec{b}^*] \exists n \geq 0.\vec{L} \longrightarrow_{\mathsf{d!}}^{n} \vec{P}(\vec{N})$

(4) $\vec{L} \in [\vec{P}(\vec{V})] \Rightarrow \exists \vec{N} \in [\vec{V}] \exists n \geq 0.\vec{L} \longrightarrow_{\mathsf{d!}}^{n} \vec{P}(\vec{N})$

PROOF. (1). We prove more generally

$$\forall \vec{L} \in [\vec{U}](C\vec{L} \in [(\vec{V}, C\vec{b}^*)]) \Rightarrow C \in [(\vec{U}, \vec{V}, C\vec{b}^*)]$$

for all $\vec{U}$ by induction on $|\vec{U}|$. In the case $\langle\rangle$, there is nothing to prove. In the case $\vec{U}, U$ assume

$$\forall \vec{L}, L \in [\vec{U}, U](C\vec{L}L \in [(\vec{V}, C\vec{b}^*)]).$$

Then for $\vec{L} \in [\vec{U}]$ we have $C\vec{L} \not\longrightarrow_{\mathsf{d}}$ and $C\vec{L}L \in [(\vec{V}, C\vec{b}^*)]$ for each $L \in [U]$. Hence $C\vec{L} \in [(U, \vec{V}, C\vec{b}^*)]$ by OINT-ARR. So the claim follows immediately from IH.

(2). Ind($\vec{P}$). The cases x, $\langle\rangle$, and $P, \vec{Q}$ are immediate.

Case $C\vec{P}$. Assume $L \longrightarrow_{\mathsf{d}}^{m} C\vec{P}(\vec{N})$. Since $\vec{P}(\vec{N}) \longrightarrow_{\mathsf{d}}^{0} \vec{P}(\vec{N})$ the IH yields for $n \geq 0$, that there exists $\vec{N}'$ such that $\vec{N} \longrightarrow_{\mathsf{d}}^{*} \vec{N}'$ and $\vec{P}(\vec{N}) \longrightarrow_{\mathsf{d!}}^{n} \vec{P}(\vec{N}')$, which implies $C\vec{P}(\vec{N}) \longrightarrow_{\mathsf{d!}}^{n} C\vec{P}(\vec{N}')$, so $L \longrightarrow_{\mathsf{d!}}^{n+m} C\vec{P}(\vec{N}')$.

(3). Ind($\vec{P}$). The cases x and $\langle\rangle$ are trivial.

Case $P, \vec{Q}$. Use IH, (2), and closure under reduction.

Case $C\vec{P}$. Recall that $C\vec{P}$ is of base type, so $L \in [C\vec{P}(\vec{b}^*)]$ implies $L \longrightarrow_{\mathsf{d}}^{m} C\vec{L}$ for some $m \geq 0$ with $\vec{L} \in [\vec{P}(\vec{b}^*)]$. By IH there exists n and $\vec{N} \in [\vec{b}^*]$ such that $\vec{L} \longrightarrow_{\mathsf{d!}}^{n} \vec{P}(\vec{N})$, so $C\vec{L} \longrightarrow_{\mathsf{d!}}^{n} C\vec{P}(\vec{N})$ by PRED-CONST-PAR, hence $\vec{L} \longrightarrow_{\mathsf{d!}}^{n+m} C\vec{P}(\vec{N})$.

(4). Use (3) together with the fact that each V_i is finite and closure under expansion and reduction. □

THEOREM 5.9 (Adequacy). $(\vec{U}, b) \in [\![\lambda\vec{x}M]\!] \Rightarrow \lambda\vec{x}M \in [(\vec{U}, b)]$.

PROOF. Induction on $(\vec{U}, b) \in [\![\lambda\vec{x}M]\!]$.

Case DEN-VAR.

$$\frac{U_i \vdash b}{(\vec{U}, b) \in [\![\lambda\vec{x}x_i]\!]}$$

We have to show that $\lambda\vec{x}x_i \in [(\vec{U}, b)]$, i.e., $\forall\vec{K} \in [\vec{U}] K_i \in [b]$. Let $\vec{K} \in [\vec{U}]$, in particular, $K_i \in [U_i]$. Thus $U_i \vdash b$ and Lemma 5.5 imply $K_i \in [b]$.

Case DEN-APP.

$$\frac{(\vec{U}, V, b) \in [\![\lambda\vec{x}M]\!] \quad (\vec{U}, V) \subseteq [\![\lambda\vec{x}N]\!]}{(\vec{U}, b) \in [\![\lambda\vec{x}.MN]\!]}$$

By IH for $\lambda\vec{x}N$ we have $\lambda\vec{x}N \in [(\vec{U}, c)]$ for each $c \in V$, so

$$\forall\vec{K} \in [\vec{U}].N[\vec{K}/\vec{x}] \in [V]. \tag{3}$$

We have to show that $\lambda\vec{x}.MN \in [(\vec{U}, b)]$, i.e.,

$$\forall\vec{K} \in [\vec{U}].(MN)[\vec{K}/\vec{x}] \in [(V, b)].$$

Let $\vec{K} \in [\vec{U}]$. The IH for $\lambda\vec{x}M$ yields $M[\vec{K}/\vec{x}] \in [(V, b)]$, which by definition implies $(M[\vec{K}/\vec{x}]L \in [b])$ for all $L \in [V]$. So the claim follows, since by (3), $N[\vec{K}/\vec{x}] \in [V]$.

Case DEN-C.

$$\frac{\vec{V} \vdash \vec{b}^*}{(\vec{U}, \vec{V}, C\vec{b}^*) \in [\![\lambda\vec{x}C]\!]}$$

Using Lemma 5.8 (1) it suffices to show that

$$\forall\vec{K} \in [\vec{U}]\forall\vec{L} \in [\vec{V}](C\vec{L} \in [C\vec{b}^*]).$$

Let $\vec{K} \in [\vec{U}]$ and $\vec{L} \in [\vec{V}]$. Thus $\vec{L} \in [\vec{b}^*]$ by monotonicity and $\vec{V} \vdash \vec{b}^*$. So by definition $C\vec{L} \in [C\vec{b}^*]$ as required.

Case DEN-D.

$$\frac{(\vec{U}, \vec{V}, b) \in [\![\lambda\vec{x}, \vec{y}M]\!] \quad D\vec{P}(\vec{y}) \rhd M \quad \vec{W} \vdash \vec{P}(\vec{V})}{(\vec{U}, \vec{W}, b) \in [\![\lambda\vec{x}D]\!]}$$

Let $\vec{L} \in [\vec{W}]$. By monotonicity we get $\vec{L} \in [\vec{P}(\vec{V})]$. We have to show $D\vec{L} \in [b]$. By Lemma 5.8 (4) there are $\vec{N} \in [\vec{V}]$ and $n \geq 0$ such that $\vec{L} \longrightarrow_{\mathsf{d!}}^{n} \vec{P}(\vec{N})$.

In particular, $D\vec{P}(\vec{N})$ is a $\mathcal{P}$-redex. Let $m \leq n$ be minimal such that there exists $\vec{K}$ with $D\vec{P}(\vec{K})$ a $\mathcal{P}$-redex and $\vec{L} \longrightarrow^m_{\mathsf{d!}} \vec{P}(\vec{K}) \longrightarrow^{n-m}_{\mathsf{d!}} \vec{P}(\vec{N})$. For such a $\vec{K}$ we also have $\vec{K} \longrightarrow^*_{\mathsf{d!}} \vec{L}$ and thus $\vec{K} \in [\vec{V}]$. Furthermore, we get by the minimality of m together with PRED-CONST-PAR that $D\vec{L} \longrightarrow^*_{\mathsf{d}} D\vec{P}(\vec{K})$, hence $D\vec{L} \longrightarrow^*_{\mathsf{d}} M[\vec{K}/\vec{y}]$. We conclude from the IH and $\vec{K} \in [\vec{V}]$ that $M[\vec{K}/\vec{y}] \in [b]$, and hence $D\vec{L} \in [b]$ by Lemma 5.5. □

6. Totality and the Density Theorem

In classical recursion theory the notion of partiality is external in the sense that one considers partial functions taking values in a codomain of total objects (usually the natural numbers). In our (higher type) setting, each $|\mathbf{C}|_\rho$ contains partial objects (e.g., $\bot := \emptyset$) and thus partiality can be seen as an internal notion. In this section we single out the total ideals among the partial continuous functionals and prove that they are dense w.r.t. the Scott topology.

6.1. Total ideals.

DEFINITION 6.1. We inductively define the set $\mathbf{G}_\rho \subseteq |\mathbf{C}_\rho|$ of *total ideals* of type ρ by:

$$\text{TOT-BASE}\ \frac{\vec{z} \in \mathbf{G}_{\vec{\tau}}}{|r_C|(\vec{z}) \in \mathbf{G}_\mu} \qquad \text{TOT-ARR}\ \frac{\forall z \in \mathbf{G}_\rho(|f|(z) \in \mathbf{G}_\sigma)}{f \in \mathbf{G}_{\rho\to\sigma}}$$

where in the first rule the constructor C has type $\vec{\tau} \to \mu$. Moreover, we inductively define a binary relation $\sim_\rho$ on total ideals of type ρ by:

$$\frac{\vec{z}_1 \sim_{\vec{\tau}} \vec{z}_2}{|r_C|(\vec{z}_1) \sim_\mu |r_C|(\vec{z}_2)} \qquad \frac{\forall z \in \mathbf{G}_\rho(|f|(z) \sim_\sigma |g|(z))}{f \sim_{\rho\to\sigma} g}$$

where again in the first rule the constructor C has type $\vec{\tau} \to \mu$. If no confusion may arise, we omit the subscripts of $\sim_\rho$ and $\mathbf{G}_\rho$.

As an immediate consequence from the definition, we note that $\sim_\rho$ is an equivalence relation on $\mathbf{G}_\rho$.

The total ideals of $|\mathbf{C_N}|$ are $\mathbf{G_N} = \{S^n0 \mid n \in \mathbb{N}\}$ and hence can be identified with $\mathbb{N}$. Notice that each $x \in \mathbf{G_N}$ is maximal but there is a maximal ideal which is not total, namely ∞. In addition, not all total ideals are maximal: the deductive closure of $\{(\{S^n0\}, 0) \mid n \in \mathbb{N}\}$ is total but not maximal because it is properly contained in the deductive closure of $\{(\emptyset, 0)\}$.

We now prove that $\sim$ is compatible with application. The proof presented here is due to Longo and Moggi [**27**]. An alternative proof uses the Density Theorem (cf. Remark 6.13).

LEMMA 6.2. *If $f \in \mathbf{G}_\rho$, $g \in |\mathbf{C}_\rho|$, and $f \subseteq g$, then $g \in \mathbf{G}_\rho$.*

PROOF. Ind($f \in \mathbf{G}_\rho$). *Case* TOT-BASE. Then $f = |r_C|(\vec{z})$ for some total $\vec{z}$. Clearly, $C\vec{*} \in f \subseteq g$, hence by Lemma 2.6 (3) there are ideals $\vec{x}$ with $g = |r_C|(\vec{x})$. Using Lemma 2.6 (1) we get $\vec{z} \subseteq \vec{x}$, so by IH $\vec{x}$ are total and therefore also $|r_C|(\vec{x}) = g$.

Case TOT-ARR. Let $z \in \mathbf{G}_\rho$, then $|f|(z) \in \mathbf{G}_\sigma$. But:

$$\begin{aligned} |f|(z) &= \{b \mid \exists U \subseteq z.(U,b) \in f\} \\ &\subseteq \{b \mid \exists U \subseteq z.(U,b) \in g\} && \text{since } f \subseteq g \\ &= |g|(z). \end{aligned}$$

Therefore by IH $|g|(z) \in \mathbf{G}_\sigma$, which is the desired conclusion. □

LEMMA 6.3. *For all $f, g \in |\mathbf{C}_{\rho\to\sigma}|$, and $x \in |\mathbf{C}_\rho|$ we have*

$$|f \cap g|(x) = |f|(x) \cap |g|(x).$$

PROOF. The following holds:

$$\begin{aligned} |f \cap g|(x) &= \{b \mid \exists U \subseteq x.(U,b) \in f \cap g\} \\ &\overset{(*)}{=} \{b \mid \exists U \subseteq x.(U,b) \in f\} \cap \{b \mid \exists U \subseteq x.(U,b) \in g\} \\ &= |f|(x) \cap |g|(x) \end{aligned}$$

where at $(*)$ the inclusion from left to right is trivial. For the converse direction, let $(U,b) \in f$ and $(V,b) \in g$ with $U, V \subseteq x$. Then also $U \cup V \subseteq x$ and thus $U \cup V$ is consistent and we have that $(U \cup V, b) \in f \cap g$ because both $\{(U,b)\}$ and $\{(V,b)\}$ entail $(U \cup V, b)$. □

LEMMA 6.4. *For all $f, g \in \mathbf{G}_\rho$, $f \sim_\rho g$ if and only if $f \cap g \in \mathbf{G}_\rho$.*

PROOF. Ind($f, g \in \mathbf{G}_\rho$). *Case* TOT-BASE. <u>$\Rightarrow$:</u> Assume $f \sim_\mu g$ then there exists a constructor C and total ideals $\vec{x}$, $\vec{y}$ with $f = |r_C|(\vec{x})$, $g = |r_C|(\vec{y})$, and $\vec{x} \sim \vec{y}$. By IH $\vec{x} \cap \vec{y}$ are total, hence also $|r_C|(\vec{x} \cap \vec{y})$. Furthermore,

$$|r_C|(\vec{x} \cap \vec{y}) \subseteq |r_C|(\vec{x}) \cap |r_C|(\vec{y})$$

and the claim follows from Lemma 6.2.

<u>$\Leftarrow$:</u> Because $f, g \in \mathbf{G}_\mu$ we have $f = |r_{C'}|(\vec{x})$ and $g = |r_{C''}|(\vec{y})$ for some total ideals $\vec{x}, \vec{y}$. Now assume $f \cap g \in \mathbf{G}_\mu$. By definition there exist total ideals $\vec{z}$ and a constructor C such that $f \cap g = |r_C|(\vec{z})$. Hence $C = C' = C''$ and by Lemma 2.6 (1) we get $\vec{z} \subseteq \vec{x} \cap \vec{y}$. Therefore by Lemma 6.2 the ideals $\vec{x} \cap \vec{y}$ are total. Now the IH yields $\vec{x} \sim \vec{y}$, so $|r_C|(\vec{x}) \sim_\mu |r_C|(\vec{y})$ as required.

Case TOT-ARR.

$$\begin{aligned} f \sim_{\rho\to\sigma} g &\leftrightarrow \forall x \in \mathbf{G}_\rho(|f|(x) \sim_\sigma |g|(x)) \\ &\leftrightarrow \forall x \in \mathbf{G}_\rho(|f|(x) \cap |g|(x) \in \mathbf{G}_\sigma) && \text{by IH} \\ &\leftrightarrow \forall x \in \mathbf{G}_\rho(|f \cap g|(x) \in \mathbf{G}_\sigma) && \text{by Lemma 6.3} \\ &\leftrightarrow f \cap g \in \mathbf{G}_{\rho\to\sigma}. \end{aligned}$$

□

THEOREM 6.5. *For all $x, y \in \mathbf{G}_\rho$, and $f \in \mathbf{G}_{\rho\to\sigma}$, $x \sim_\rho y$ implies $|f|(x) \sim_\sigma |f|(y)$.*

PROOF. If $x \sim_\rho y$, then $x \cap y \in \mathbf{G}_\rho$ by Lemma 6.4 and hence $|f|(x \cap y) \in \mathbf{G}_\sigma$. Because of $|f|(x \cap y) \subseteq |f|(x) \cap |f|(y)$ we obtain $|f|(x) \cap |f|(y) \in \mathbf{G}_\sigma$. Again using Lemma 6.4, we conclude $|f|(x) \sim_\sigma |f|(y)$. □

6.2. The Density Theorem. The Density Theorem says that $\mathbf{G}_\rho \subseteq |\mathbf{C}_\rho|$ is dense w.r.t. the Scott topology on $|\mathbf{C}_\rho|$. Equivalently, this can be stated as

$$\forall U \in \mathrm{Con}_\rho \, \exists x \in \mathbf{G}_\rho \, U \subseteq x.$$

Clearly, this requires that each $\mathbf{G}_\rho$ is non-empty. In general, this is not satisfied for all types (e.g., consider the type $\mu\alpha(\alpha \to \alpha)$). Therefore we assume in the remaining part of this chapter that all base types are *inhabited*. More precisely, we require for all simultaneously defined algebras $\vec{\mu} = \mu\vec{\alpha}.\vec{\kappa}$ and α_j that there exists a constructor type $\kappa \in \vec{\kappa}$ with

$$\kappa = \vec{\rho} \to (\vec{\sigma}_1 \to \alpha_{j_1}) \to \cdots \to (\vec{\sigma}_n \to \alpha_{j_n}) \to \alpha_j$$

and for all α_{j_i} we have $j_i < j$. It is easily seen by induction on the type that for each type there is a closed term of that type whose denotation is total and therefore each $\mathbf{G}_\rho$ is inhabited.

The Density Theorem is due to Kreisel [**26**]; our proof is based on Berger's [**3**] and incorporates ideas from Schwichtenberg [**36**]. We improve on [**36**] by allowing arbitrary (inhabited) base types.

Definition 6.6. We simultaneously define the *depth* of an extended token and of a formal neighborhood by

$$\mathrm{dp}(\emptyset) := \mathrm{dp}(*) := 0, \; \mathrm{dp}(U) := \max\{\mathrm{dp}(a) \mid a \in U\} \text{ for } U \neq \emptyset,$$
$$\mathrm{dp}(C\vec{a}^*) := \max \mathrm{dp}(\vec{a}^*) + 1, \text{ and } \mathrm{dp}((U,a)) := \max\{\mathrm{dp}(U), \mathrm{dp}(a)\} + 1.$$

Remark 6.7. If $U \in \mathrm{Con}_\mu$ is non-empty, there exists a constructor C such that $C\vec{U} \vdash U$ and $\mathrm{dp}(\vec{U}) < \mathrm{dp}(U)$, where $U_i := \{a \mid \exists C\vec{a}^* \in U . a = a_i^*\}$ and $C\vec{U} := (C\vec{x})[\vec{U}/\vec{x}]$. Note that we also have $U \vdash C\vec{U}$. (Cf. the proof of Lemma 3.5 (1).)

We extend inconsistency to arbitrary sets of tokens x and y by

$$x \not\smile y :\leftrightarrow \exists a \in x \exists b \in y \, a \not\smile b.$$

Lemma 6.8. *For all* $U, V \in \mathrm{Con}_\rho$*:*

(i) $\exists x \in \mathbf{G}_\rho (U \subseteq x)$, *and*
(ii) $U \not\smile V \to \exists z \in \mathbf{G}_{\rho\to\mathbf{B}} \, z(\overline{U}) \not\smile z(\overline{V})$.

Proof. Induction on $\max\{\mathrm{dp}(U), \mathrm{dp}(V)\}$ with a case distinction on the type ρ. *Case* μ. If $U = \emptyset$, both statements are trivial (here we need the assumption that each base type has total ideals). So assume $U \neq \emptyset$. By the remark above, there is a constructor C and $\vec{U}$ with $C\vec{U} \vdash U$, $\mathrm{dp}(\vec{U}) < \mathrm{dp}(U)$, and the U_i are the corresponding components of tokens in U.

(i). By IH (i) for $\vec{U}$, there are total ideals $\vec{x}$ with $\vec{U} \subseteq \vec{x}$. For $x := |r_C|(\vec{x}) \in \mathbf{G}_\mu$, we have $U \subseteq x$ because $C\vec{U} \subseteq x$ and $C\vec{U} \vdash U$.

(ii). Assume $U \not\smile V$. Note that $V \neq \emptyset$, and therefore by the remark above there is a constructor C' and $\vec{V}$ with $C'\vec{V} \vdash V$, $\mathrm{dp}(\vec{V}) < \mathrm{dp}(V)$, and the V_i are the corresponding components of tokens in V. Then also $C\vec{U} \not\smile C'\vec{V}$.

Subcase $C' \neq C$. Using computation rules, it is easy to see that there is a total $z \in \mathbf{G}_{\rho\to\mathbf{B}}$ with $z(C\vec{x}) = \mathtt{tt}$ and $z(C''\vec{y}) = \mathtt{ff}$ for each constructor C'' of μ different from C. Then $z(\overline{U}) = \mathtt{tt} \not\smile \mathtt{ff} = z(\overline{V})$.

Subcase $C' = C$. Then there is some i with $U_i \not\smile_\tau V_i$. By IH (ii) there exists $z' \in \mathbf{G}_{\tau\to\mathbf{B}}$ with $z'(\overline{U}_i) \not\smile z'(\overline{V}_i)$. Clearly, there is a total $p \in \mathbf{G}_{\mu\to\tau}$ satisfying $p(C\vec{x}) = x_i$ and $p(C''\vec{y}) = y$ for each constructor $C'' \neq C$ of μ where y is an arbitrary but fixed total ideal of type τ. Then $z := z' \circ p$ is total and has the desired property.

Case $\rho \to \sigma$. (ii). Let $W_1, W_2 \in \mathrm{Con}_{\rho\to\sigma}$ and suppose $W_1 \not\smile W_2$. Then there are $(U_i, a_i) \in W_i$ $(i = 1, 2)$ with $U_1 \smile U_2$ and $a_1 \not\smile a_2$. Because

$$\mathrm{dp}(U_1 \cup U_2) < \max\{\mathrm{dp}(W_1), \mathrm{dp}(W_2)\}$$

the IH (i) yields $x \in \mathbf{G}_\rho$ with $U_1 \cup U_2 \subseteq x$. Since $\overline{W}_1(x) \ni a_1 \not\smile a_2 \in \overline{W}_2(x)$ and IH (ii) there is a total $z' \in \mathbf{G}_{\sigma\to\mathbf{B}}$ with $z'(\overline{\{a_1\}}) \not\smile z'(\overline{\{a_2\}})$. Hence also $z'(\overline{W}_1(x)) \not\smile z'(\overline{W}_2(x))$, so take $z \in \mathbf{G}_{(\rho\to\sigma)\to\mathbf{B}}$ defined by $z(y) := z'(yx)$.

(i). Let $W = \{(U_i, a_i) \mid i \in I\} \in \mathrm{Con}_{\rho\to\sigma}$. For $i, j \in I$ with $a_i \not\smile a_j$, we also have $U_i \not\smile U_j$. By IH (ii) there exists $z_{ij} \in G_{\rho\to\mathbf{B}}$ such that

(4) $$k_{ij} := z_{ij}(\overline{U}_i) \not\smile z_{ij}(\overline{U}_j) =: l_{ij}.$$

We may choose $z_{ij} = z_{ji}$, thus also $k_{ij} = l_{ji}$ and $l_{ij} = k_{ji}$. Since $k_{ij} \not\smile k_{ji}$ we have $\{k_{ij}, k_{ji}\} = \{\mathtt{tt}, \mathtt{ff}\} = \mathbf{G}_B$. Hence all k_{ij}'s are maximal and finite ideals.

Given $U \in \mathrm{Con}_\rho$ we define

$$I_U := \{i \in I \mid \forall j \in I(a_i \not\smile a_j \to z_{ij}(\overline{U}) \supseteq k_{ij})\}$$

and $W_U := \{a_i \mid i \in I_U\}$. We first prove that $W_U \in \mathrm{Con}_\sigma$. Let $i, j \in I_U$ and suppose $a_i \not\smile a_j$. Because $i \in I_U$ we have $z_{ij}(\overline{U}) \supseteq k_{ij}$, and because $j \in I_U$ we conclude $z_{ji}(\overline{U}) \supseteq k_{ji}$. But $z_{ij} = z_{ji}$, so we obtain that $z_{ij}(\overline{U})$ is inconsistent by (4), a contradiction. Hence $a_i \smile a_j$. This completes the proof of $W_U \in \mathrm{Con}_\sigma$.

Note that $\mathrm{dp}(W_U) < \mathrm{dp}(W)$, thus by IH (i) there exists $y_U \in \mathbf{G}_\sigma$ with $W_U \subseteq y_U$. Define the relation $r \subseteq \mathrm{Con}_\rho \times C_\sigma$ by

$$r(U, a) :\leftrightarrow \begin{cases} a \in y_U & \text{if } \forall i, j(a_i \not\smile a_j \to z_{ij}(\overline{U}) \text{ is total}), \\ W_U \vdash a & \text{otherwise.} \end{cases}$$

We will now show that r is an approximable map which is total and extends W.

For $W \subseteq r$ we have to show $r(U_i, a_i)$ for all $i \in I$. By definition $z_{ij}(\overline{U}_i) = k_{ij}$ and hence $i \in I_{U_i}$ which clearly yields $r(U_i, a_i)$.

We now prove that

(5) $$r(U, b_1) \wedge r(U, b_2) \to b_1 \smile b_2.$$

So assume the premises. If $z_{ij}(\overline{U})$ is total for all i and j with $a_i \not\smile a_j$, then $b_1, b_2 \in y_U$. So $b_1 \smile b_2$ because y_U is an ideal. Otherwise, we have $W_U \vdash b_1, b_2$, hence $b_1 \smile b_2$, which concludes the proof of (5).

Next, we prove

(6) $$V_1 \vdash V_2 \wedge r(V_2, U) \wedge U \vdash b \to r(V_1, b).$$

Assume the premises. Because of $V_1 \vdash V_2$ we obtain $z_{ij}(\overline{V}_1) \supseteq z_{ij}(\overline{V}_2)$ so in particular, $I_{V_2} \subseteq I_{V_1}$. If all $z_{ij}(\overline{V}_2)$ are total, then so are all $z_{ij}(\overline{V}_1) \supseteq z_{ij}(\overline{V}_2)$. Because each ideal in $\mathbf{G}_\mathbf{B}$ is maximal we obtain $z_{ij}(\overline{V}_1) = z_{ij}(\overline{V}_2)$ and hence $I_{V_2} = I_{V_1}$. In this case, $r(V_2, U)$ yields $U \subseteq y_{V_2}$. So $U \subseteq y_{V_2} = y_{V_1}$

and hence $b \in y_{V_1}$, i.e., $r(V_1, b)$. In the other case we have $W_{V_2} \vdash U \vdash b$, but $W_{V_1} \supseteq W_{V_2}$ so $r(V_1, b)$ in any case. This finishes the proof of (6).

By (5) and (6), r is an approximable map. It remains to prove that r is total, i.e., $r \in \mathbf{G}_{\rho\to\sigma}$. Let $x \in \mathbf{G}_\rho$. We have to show $|r|(x) \in \mathbf{G}_\sigma$. Since x is total, we have that all $z_{ij}(x)$'s are total. But $z_{ij}(x) \in \mathbf{G}_\mathbf{B}$ is finite, so there exists $U_{ij} \subseteq x$ such that $z_{ij}(x) = z_{ij}(\overline{U}_{ij})$. Let U be the union of all U_{ij} ($i, j \in I$ with $a_i \not\smile a_j$). Then $U \subseteq x$ and $U \in \mathrm{Con}_\sigma$. By the monotonicity of the z_{ij}'s, we conclude that all $z_{ij}(\overline{U})$'s are total. Hence $r(U, b)$ for all $b \in y_U$. We get $y_U \subseteq |r|(x)$, so by Lemma 6.2 $|r|(x) \in \mathbf{G}_\sigma$ since $y_U \in \mathbf{G}_\sigma$. This completes the proof. □

Corollary 6.9 (Density Theorem). *The total ideals $\mathbf{G}_\rho$ are dense in $|\mathbf{C}_\rho|$.*

Corollary 6.10. *Each type ρ is separable, i.e.,*

$$\forall U, V \in \mathrm{Con}_\rho \left(U \not\smile V \to \exists z \in \mathbf{G}_{\rho\to\mathbf{B}}\, z(\overline{U}) \not\smile z(\overline{V}) \right).$$

Corollary 6.11. *For all $x, y \in \mathbf{G}_\rho$:*

$$x \sim_\rho y \leftrightarrow x \cup y \text{ consistent.}$$

Proof. $\underline{\Rightarrow:}$ $\mathrm{Ind}(x \sim y)$. We only treat the case

$$\frac{\forall z \in \mathbf{G}_\rho\, f(z) \sim_\sigma g(z)}{f \sim_{\rho\to\sigma} g}.$$

Let $(U, a) \in f$ and $(V, b) \in g$ with $U \smile V$. We have to show $a \smile b$. By the density theorem, there is a total ideal $z \in \mathbf{G}_\rho$ with $U \cup V \subseteq z$. By the IH, $f(z) \cup g(z)$ is consistent, and therefore $a \smile b$ because $a, b \in f(z) \cup g(z)$.

$\underline{\Leftarrow:}$ Let $x \cup y$ be consistent, then $z := \overline{(x \cup y)}$ is an ideal. But z extends $x \in \mathbf{G}$, so z is total as well. Moreover, $x \cap z = x \in \mathbf{G}$ and $y \cap z = y \in \mathbf{G}$ and therefore by Lemma 6.4, $x \sim z \sim y$.[2] □

Remark 6.12. The expert is invited to inspect that the proof of Lemma 6.8 also gives rise to effective versions of density and separability in the following sense. Given an effective coding $\ulcorner \cdot \urcorner$ of tokens and neighborhoods as natural numbers we call an ideal *computable* if the (codes of) its tokens are Σ^0_1-definable. The Effective Density Theorem states that for each type ρ there is a total and computable ideal x_ρ such that $U \subseteq x_\rho(\ulcorner U \urcorner)$ for all $U \in \mathrm{Con}_\rho$. Analogously, effective separability states the existence of an total and computable ideal z_ρ such that $z_\rho(\ulcorner U \urcorner, \ulcorner V \urcorner, \overline{U}) \not\smile_\mathbf{B} z_\rho(\ulcorner U \urcorner, \ulcorner V \urcorner, \overline{V})$ whenever $U \not\smile_\rho V$. Moreover, the map which assigns a code $\ulcorner \rho \urcorner$ of a type ρ to an enumeration of (the codes of) the ideal x_ρ (respectively z_ρ) is computable.

Remark 6.13. Using the characterization from Corollary 6.11 we can give an alternative proof of theorem 6.5. Given total x and y with $x \sim y$, we obtain that $\overline{x \cup y}$ is a total ideal by the corollary. Thus for all total f, $f(\overline{x \cup y})$ is total and hence $f(x) \cup f(y) \subseteq f(\overline{x \cup y})$ is consistent. Again with the corollary we conclude $f(x) \sim f(y)$.

[2] An alternative proof of "$\Leftarrow$" by induction on $x, y \in \mathbf{G}_\rho$ does not need Lemma 6.4.

7. Kleene-Kreisel Continuous Functionals

The continuous functionals were independently introduced by Kleene [**23**] and Kreisel [**26**]. There are several ways to describe the Kleene-Kreisel continuous functionals (see, for example, [**31**, p. 253 f.] for an overview and further references). Here we follow the approach of Ershov [**12**] and introduce the continuous functionals as $\sim$-equivalence classes of total ideals.

DEFINITION 7.1. The *Kleene-Kreisel continuous functionals* of type ρ are defined as the equivalence classes of $\sim_\rho$, i.e., $\mathbb{G}_\rho := \mathbf{G}_\rho / \sim_\rho$. We denote the equivalence class of $f \in \mathbf{G}_\rho$ w.r.t. $\sim_\rho$ by $[f]_\rho$ or sometimes just by $[f]$.

For $[f] \in \mathbb{G}_{\rho\to\sigma}$ we can associate a function $\mathbb{G}([f])\colon \mathbb{G}_\rho \to \mathbb{G}_\sigma$ given by $\mathbb{G}([f])([x]) := [|f|(x)]$. This is well-defined since whenever $x \sim_\rho y$ and $f \sim_{\rho\to\sigma} g$, then $|f|(x) \sim_\sigma |g|(x)$ and by Theorem 6.5 $|g|(x) \sim_\sigma |g|(y)$. Therefore $|f|(x) \sim_\sigma |g|(y)$.

Note that, by Corollary 6.11, each equivalence class $[x]$ possesses a unique maximal representative, namely $\overline{\bigcup[x]}$ (which is equal to $\bigcup[x]$ because $[x]$ is directed by the corollary). We call this the *canonical representative* of $[x]$.

Next, we introduce (classical) models over the continuous functionals. These will be models where our recursion schemes studied in Chapter 3 are valid. To prove this, we need a way to reason about non-total functionals as well. This is done via set variables.

There are two ways how to handle equality. On the one hand, it can be interpreted as equality between ideals resulting in the model $\mathbf{G}$. On the other hand, equality can be interpreted as the relation $\sim$ resulting in the extensional model $\mathbb{G}$.

We first need to know that the denotation of a $T_\mathcal{B}$-term is total.

LEMMA 7.2. *Let M be a term of $T_\mathcal{B}$ and let θ be an environment of ideals (cf. Definition 4.1) such that $\theta(x)$ is total for all $x \in \mathsf{FV}(M)$. Then $[\![M]\!]_\theta$ is total.*

PROOF SKETCH. First one proves that all recursion operators $\mathcal{R}_\mu$ have a total denotation. This is done by showing that $[\![\mathcal{R}_\mu]\!](x)$ is total by induction on $x \in \mathbf{G}_\mu$.

Then the claim follows easily by induction on the buildup of the term M using the results from Section 4. $\square$

We now define the validity of a $\mathsf{HA}^\omega[\mathcal{X}]$-formula in $\mathbf{G}$ and $\mathbb{G}$. This is done as usual and we handle the predicate variables in $\mathcal{X}$ as set variables via an assignment.

DEFINITION 7.3. An *environment of total ideals* θ is an environment of ideals θ such that $\theta(x)$ is total for all $x \in \mathsf{supp}(\theta)$. An $\mathcal{X}$-assignment is map Ξ with domain $\mathcal{X}$ such that

$$\Xi(X) \subseteq \mathbf{G}_{\rho_1} \times \cdots \times \mathbf{G}_{\rho_n}$$

for each $X \in \mathcal{X}$ with $\mathsf{ar}(X) = (\rho_1, \ldots, \rho_n)$. Moreover, Ξ is $\sim$*-compatible* if in addition

$$\vec{x} \in \Xi(X) \land \vec{x} \sim \vec{y} \to \vec{y} \in \Xi(X)$$

for all $X \in \mathcal{X}$.

Now given an environment of total ideals θ and an $\mathcal{X}$-assignment Ξ we define $\mathbf{G}, \Xi \models A[\theta]$ for all $\mathsf{HA}^\omega[\mathcal{X}]$-formulas A with $\mathsf{FV}(A) \subseteq \mathsf{supp}(\theta)$:[3]

$$\begin{aligned}
\mathbf{G}, \Xi &\models M =_\rho N[\theta] && :\leftrightarrow [\![M]\!]_\theta = [\![N]\!]_\theta \\
\mathbf{G}, \Xi &\models X(\vec{M})[\theta] && :\leftrightarrow [\![\vec{M}]\!]_\theta \in \Xi(X) \\
\mathbf{G}, \Xi &\models \bot && :\leftrightarrow \bot \\
\mathbf{G}, \Xi &\models (A \circ B)[\theta] && :\leftrightarrow (\mathbf{G}, \Xi \models A[\theta]) \circ (\mathbf{G}, \Xi \models B[\theta]) \\
\mathbf{G}, \Xi &\models (\mathsf{Q} x^\rho A)[\theta] && :\leftrightarrow \mathsf{Q} z \in \mathbf{G}_\rho\, \mathbf{G}, \Xi \models A[\theta[x \mapsto z]]
\end{aligned}$$

for $\circ = \vee, \wedge, \to$ and $\mathsf{Q} = \forall, \exists$. $\mathbb{G}, \Xi \models A[\theta]$ is defined analogously to $\mathbf{G}, \Xi \models A[\theta]$ but the equality relation is interpreted as $\sim$, i.e.,

$$\mathbb{G}, \Xi \models M =_\rho N[\theta] \;:\leftrightarrow\; [\![M]\!]_\theta \sim_\rho [\![N]\!]_\theta.$$

We write $\mathbf{G} \models A[\theta]$ ($\mathbb{G} \models A[\theta]$) if $\mathbf{G}, \Xi \models A[\theta]$ ($\mathbb{G}, \Xi \models A[\theta]$) holds for all ($\sim$-compatible) Ξ. Moreover, we omit θ if A is closed.

Remark 7.4. (1) The model $\mathbb{G}$ validates extensionality, whereas the model $\mathbf{G}$ doesn't: there are total functions which agree on all total arguments but are different on non-total arguments (e.g., $\overline{\{(\emptyset, 0)\}}$ and $\overline{\{(\{S^n 0\}, 0) \mid n \in \mathbb{N}\}}$).

(2) Let Ξ be a $\sim$-compatible $\mathcal{X}$-assignment and A (B) be a $\mathsf{HA}[\mathcal{X}]$-formula such that for each type ρ, whenever $=_\rho$ occurs positively (negatively) in A (B), then ρ is a finitary base type. Then the following *transfer principles* hold:

$$\begin{aligned}
&\mathbb{G}, \Xi \models A[\theta] \text{ implies } \mathbf{G}, \Xi \models A[\theta], \text{ and} \\
&\mathbf{G}, \Xi \models B[\theta] \text{ implies } \mathbb{G}, \Xi \models B[\theta].
\end{aligned}$$

To see this, observe that $\sim$ is reflexive and for a finitary base type μ, we have $x \sim_\mu y$ iff $x = y$, for all total x and y. The rest follows by induction on formulas.

Theorem 7.5. *$\mathbb{G} \models \mathsf{E{-}PA}^\omega[\mathcal{X}]$ and $\mathbf{G} \models \mathsf{PA}^\omega[\mathcal{X}]$.*

Proof sketch. The induction axioms are validated using induction on the definition of the $\mathbf{G}_\rho$'s. To verify the compatibility axioms in the case of $\mathbb{G}$ one needs the compatibility property of the $\mathcal{X}$-assignments. The rest of the proof is straightforward. □

7.1. Classical Analysis. We will now show that the Kleene-Kreisel continuous functionals verify the axiom of countable and dependent choice and thus are a model of classical analysis.

Axioms of Choice. The *axiom of dependent choice* is the following axiom scheme

$$\mathrm{DC}_\rho \qquad \forall n^{\mathbf{N}} \forall x^\rho \exists y^\rho A(n, x, y) \to \exists f^{\mathbf{N} \to \rho} \forall n^{\mathbf{N}} A(n, f(n), f(n+1)),$$

[3] Of course, the connectives on the right hand side are from the metalanguage, whereas on the left hand side they are from the object language.

where A ranges over $\mathsf{HA}[\mathcal{X}]$-predicates with $f \notin \mathsf{FV}(A)$. With DC we denote the union of all DC_ρ. The *axiom of (countable) choice* is given by the scheme

$$\mathrm{AC}_\rho \qquad \forall n^{\mathbf{N}} \exists x^\rho A(n,x) \to \exists f^{\mathbf{N}\to\rho} \forall n^{\mathbf{N}} A(n, f(n)).$$

Again, $A(n,x)$ ranges over $\mathsf{HA}^\omega[\mathcal{X}]$-predicates with $f \notin \mathsf{FV}(A)$, and AC stands for the union of all AC_ρ. Notice that this is a special case of the *general axiom of choice* given by

$$\mathrm{GAC}_{\rho,\sigma} \qquad \forall x^\rho \exists y^\sigma A(x,y) \to \exists f^{\rho\to\sigma} \forall x^\rho A(x, f(x)).$$

Since we are mainly interested in the axiom of countable choice AC instead of the more general GAC we always mean the former when referring to "the axiom of choice".

We have suppressed the reference to $\mathcal{X}$ in the notation of DC, AC, and GAC. No confusion should arise since we need the predicate variables $\mathcal{X}$ only for the totality proofs in the next chapter — otherwise we explicitly set $\mathcal{X}$ to be empty.

It is easy to see that we can identify $\mathbf{G}_\mathbf{N}$ as well as $\mathbb{G}_\mathbf{N}$ with $\mathbb{N}$.

LEMMA 7.6. *The elements of* $\mathbb{G}_{\mathbf{N}\to\rho}$ *correspond to* $\mathbb{G}_\rho$*-sequences in the following way:*

$$\{\mathbb{G}(\alpha) \mid \alpha \in \mathbb{G}_{\mathbf{N}\to\rho}\} = (\mathbb{G}_\rho)^\mathbb{N}.$$

PROOF. We have already seen that given $\alpha \in \mathbb{G}_{\mathbf{N}\to\rho}$, $\mathbb{G}(\alpha)$ is a map from $\mathbb{G}_\mathbf{N} = \mathbb{N}$ to $\mathbb{G}_\rho$. Conversely, let $f \colon \mathbb{N} \to \mathbb{G}_\rho$. We have to show $f = \mathbb{G}(\alpha)$ for some $\alpha \in \mathbb{G}_{\mathbf{N}\to\rho}$. Define $g \colon \mathbb{N} \to \mathbf{G}_\rho$ by setting $g(n)$ to be the canonical representative of $f(n)$, so in particular $f(n) = [g(n)]$ for all $n \in \mathbb{N}$. We now define $h \colon |\mathbf{C}_\mathbf{N}| \to |\mathbf{C}_\rho|$ by

$$h(x) = \begin{cases} g(x) & \text{if } x \in \mathbf{G}_\mathbf{N}, \\ \emptyset & \text{otherwise.} \end{cases}$$

It is not hard to see that h is continuous. Therefore $\widehat{h} \in |{\mathbf{C}_\rho}^{\mathbf{C}_\mathbf{N}}|$ and since h agrees with g on total arguments we have $\widehat{h} \in \mathbf{G}_{\mathbf{N}\to\rho}$. So we get with $\alpha := [\widehat{h}] \in \mathbb{G}_{\mathbf{N}\to\rho}$ for $n \in \mathbb{N}$:

$$\mathbb{G}(\alpha)(n) = [|\widehat{h}|(n)] = [h(n)] = [g(n)] = f(n). \qquad \square$$

THEOREM 7.7. $\mathbb{G} \models \mathrm{DC}$ *and* $\mathbf{G} \models \mathrm{DC}$.

PROOF. The proof is similar for both. For example, assume

$$\mathbb{G}, \Xi \models \forall n \forall x^\rho \exists y^\rho A(n,x,y)[\theta], \text{ i.e.,}$$

$$\forall k \in \mathbb{N} \forall x' \in \mathbf{G}_\rho \exists y' \in \mathbf{G}_\rho.\ \mathbb{G}, \Xi \models A(n,x,y)[\theta[n,x,y \mapsto k, x', y']]$$

By dependent choice in the metatheory, there exists a function $g \colon \mathbb{N} \to \mathbf{G}_\rho$ such that

$$\forall k \in \mathbb{N}.\ \mathbb{G}, \Xi \models A(n,x,y)[\theta[n,x,y \mapsto k, g(k), g(k+1)]] \tag{7}$$

As in the lemma above, we can extend g to a continuous $h \colon |\mathbf{C}_\mathbf{N}| \to |\mathbf{C}_\rho|$ with $\widehat{h} \in \mathbf{G}_{\mathbf{N}\to\rho}$. From (7), we obtain

$$\forall k \in \mathbb{N}.\ \mathbb{G}, \Xi \models A(n, f(n), f(n+1))[\theta[n, f \mapsto k, \widehat{h}]], \text{ i.e.,}$$

$$\mathbb{G}, \Xi \models \exists f \forall n A(n, f(n), f(n+1))[\theta] \qquad \square$$

It is not hard to see that AC is provable from DC and thus is also valid in our models.

7.2. Continuity. We now assume that our types are closed under list types (i.e., if $\rho \in \mathcal{B}$ then so is $\rho^* \in \mathcal{B}$) and contain $\mathbf{N}$. Using the recursion operator for lists, we can define the initial segment $\bar{\alpha}n$ of $\alpha\colon \mathbf{N} \to \rho$ such that the following is provable in HA^ω:

$$\bar{\alpha}0 = \langle\rangle \qquad \bar{\alpha}(n+1) = \bar{\alpha}n * \alpha(n).$$

The *continuity axiom* is given by[4]

$$\mathrm{Cont}_\rho \quad \forall Y^{(\mathbf{N}\to\rho)\to\mathbf{N}} \forall\alpha\exists n\forall\beta(\bar{\alpha}n =_{\rho^*} \bar{\beta}n \to Y\alpha =_{\mathbf{N}} Y\beta).$$

We say that n is a *point of continuity* of Y at α expressing that

$$\forall\beta(\bar{\alpha}n = \bar{\beta}n \to Y(\alpha) = Y(\beta)).$$

Our next goal is to show that the continuity axiom Cont holds in the model $\mathbb{G}$. For this we first need some preparations. Let $\alpha \in |\mathbf{C}_{\mathbf{N}\to\rho}|$ be a sequence and define $\alpha_\perp$ by

$$(U, a) \in \alpha_\perp :\leftrightarrow (U, a) \in \alpha \wedge \exists k \in \mathbb{N}\, S^k 0 \in U. \tag{8}$$

Then $\alpha_\perp \subseteq \alpha$ so $\alpha_\perp$ is consistent. It is also deductively closed: suppose $W \subseteq \alpha_\perp$ with $W \vdash (U, a)$. We show $(U, a) \in \alpha_\perp$. Because α is deductively closed $(U, a) \in \alpha$. By Lemma 2.7, we get $W(U) \neq \emptyset$ so there is $(V, b) \in W$ with $U \vdash V$. Because $(V, b) \in \alpha_\perp$ there exists $k \in \mathbb{N}$ with $U \vdash S^k 0$ and therefore $S^k 0 \in U$. This proves $(U, a) \in \alpha_\perp$.

It is not hard to see that $\alpha_\perp(n) = \alpha(n)$ for all $n \in \mathbb{N}$. In particular, if α is total, then so is $\alpha_\perp$ and $\alpha_\perp \sim \alpha$.

Theorem 7.8. $\mathbb{G} \models \mathrm{Cont}$.

Proof. Let Y and α be total. By the above considerations, w.l.o.g. we can assume $\alpha = \alpha_\perp$. Because Y and α are total, so is $Y(\alpha)$, say $Y(\alpha) = m \in \mathbb{N}$. Hence there exists $U \subseteq \alpha$ such that $(U, S^m 0) \in Y$. Define

$$n := \sup\{i \mid \exists (W, c) \in U.\, S^i 0 \in W\} + 1 \in \mathbb{N}.$$

It remains to prove

$$\forall\beta \in \mathbf{G}_{\mathbf{N}\to\rho} \left(\bar{\alpha}n \sim \bar{\beta}n \to Y(\beta) = m\right)$$

where we have written $\bar{\alpha}n$ instead of $[\![\lambda xy.\bar{x}y]\!](\alpha, n)$ and similarly for $\bar{\beta}n$. Let $\beta \in \mathbf{G}_{\mathbf{N}\to\rho}$ with $\bar{\alpha}n \sim \bar{\beta}n$. Because $\beta_\perp \sim \beta$, w.l.o.g. we can assume $\beta = \beta_\perp$. Since Y and β are total, $Y(\beta) = k \in \mathbb{N}$ is total, so there exists $V \subseteq \beta$ with $(V, S^k 0) \in Y$. We have to show $k = m$. For this, it suffices to show $S^m 0 \smile S^k 0$. Because Y is consistent, this amounts to proving $U \smile V$. Let $(W_1, a_1) \in U$ and $(W_2, a_2) \in V$ with $W_1 \smile W_2$ be given. We show: $a_1 \smile a_2$. Since $W_1 \smile W_2$, $\alpha = \alpha_\perp$, and $\beta = \beta_\perp$, there exists $l \in \mathbb{N}$ with $S^l 0 \in W_1 \cap W_2$. Because $(W_1, a_1) \in U$, we conclude $l < n$ and therefore $\alpha l \sim \beta l$ by assumption. Corollary 6.11 implies that $\alpha l \cup \beta l$ is consistent. Clearly, $a_1, a_2 \in \alpha l \cup \beta l$ and thus $a_1 \smile a_2$. □

[4]This in fact reflects topological continuity. The space of $X^{\mathbb{N}}$ of sequences of a set X carries a natural topology: The basic open sets are given by the sets of all sequences sharing some initial segment. The continuity principle above reflects topological continuity when $\mathbb{N}$ is equipped with the discrete topology.

Together with Remark 7.4 we obtain:

COROLLARY 7.9. $\mathbf{G} \models \mathrm{Cont}$.

8. Notes

The material of Sections 1–5 is based on [**36**] and earlier versions of [**37**]. Apart from some corrections, the main difference in our treatment is the use of coherent information systems instead of atomic coherent information systems (acis's). We now elaborate on why we have done so.

A c.i.s. is *atomic*, if for each formal neighborhood U and token a, $U \vdash a$ iff there is a $b \in U$ with $\{b\} \vdash a$. Note that our c.i.s.'s $\mathbf{C}_\rho$ are in general not atomic. E.g., consider an algebra with a binary constructor B and a nullary constructor 0, then $\{B0{*}, B{*}0\} \vdash B00$. But neither $\{B0{*}\}$ nor $\{B{*}0\}$ alone entail $B00$.

For each type ρ one can define an acis $\mathbf{C}'_\rho$. Loosely speaking, this can be done as in Definition 2.2 with the following modifications. For entailment one restricts the rule for base types to $n = 1$, for arrow types one uses $\{(U, a)\} \vdash' (V, b)$ defined as $V \vdash' U$ and $\{a\} \vdash' b$ instead, and adds the rule $U \vdash' a$ whenever there is a $b \in U$ with $\{b\} \vdash' a$. The tokens are defined as for the non-atomic case, but constructors take formal neighborhoods at non-finitary argument positions instead of extended tokens. We refer the reader to [**36**] for the detailed definition. As for $|\mathbf{C}_\rho|$, one can define the total ideals of $|\mathbf{C}'_\rho|$, say $\mathbf{G}'_\rho$. It turns out that the corresponding primed version of Lemma 6.2 does not hold.

As a counterexample consider an algebra μ with a constructor C of type $(\mathbf{N} \to \mathbf{N}) \to (\mathbf{N} \to \mathbf{N}) \to \mu$. Let x be the deductive closure of $\{(\{S^n 0\}, 0) \mid n \in \mathbb{N}\}$ w.r.t. $\mathbf{C}'_\rho$, and let $U := \{(\emptyset, 0)\}$. Then x is in $\mathbf{G}'$. Now define $f := |r_C|(x, x) \in \mathbf{G}'$ and $g := f \cup \overline{\{CU\emptyset\}} \cup \overline{\{C\emptyset U\}}$ where the deductive closure is w.r.t. $\mathbf{C}'_\rho$. Because of the atomicity, g is deductively closed. It is not hard to see that g is in fact an ideal and that $CUU \notin g$. Hence g is not of the form $|r_C|(x_1, x_2)$ for some ideals x_1, x_2. In particular, g is not in $\mathbf{G}'$. Therefore, we have found an ideal g, which extends a total ideal f, but is *not* total itself.

This also means that Lemma 2.6 (3) does not hold in the acis setting: There are non-empty ideals $z \in |\mathbf{C}'_\mu|$ which are not of the form $z = |r_C|(\vec{x})$ for some ideals $\vec{x}$ and a constructor C. Take, for instance, the ideal g from above, or the ideal $\{B0{*}, B{*}0\} \in |\mathbf{C}'|$ with B and 0 as above.

However, if one considers base types with only nullary and unary constructors, these problems vanish and in this case the approach with acis's is more perspicuous.

CHAPTER 3

Recursion

In this chapter we study recursion schemes which can be used to give a computational interpretation of various choice principles. We will show that most of the schemes are valid in the Kleene-Kreisel continuous functionals and thus consistent. This will be done in two steps. First, we show that the schemes exist as partial continuous functionals. Then, we show that these partial continuous functionals are total. For the totality proofs we need axiom schemes which naturally correspond to the recursion schemes. Another question we tackle in this chapter is the interdefinability of the studied functionals. Moreover, we show that the fan functional is definable by two of our schemes.

Notational Conventions. From now on we restrict our attention to a fixed system of types given by the grammar

$$\rho, \sigma ::= \mathbf{N} \mid \rho^* \mid \rho \times \sigma \mid \rho \to \sigma.$$

We set $\rho^\omega := \mathbf{N} \to \rho$ for the type of sequences of objects of type ρ. The *level* $\mathrm{lev}(\rho)$ of a type ρ is inductively defined by

$$\mathrm{lev}(\mathbf{N}) := 0, \qquad \mathrm{lev}(\rho \times \sigma) := \max\{\mathrm{lev}(\rho), \mathrm{lev}(\sigma)\},$$
$$\mathrm{lev}(\rho^*) := \mathrm{lev}(\rho), \qquad \mathrm{lev}(\rho \to \sigma) := \max\{\mathrm{lev}(\rho) + 1, \mathrm{lev}(\sigma)\}.$$

By term we always mean a term of HA^ω with the above system of types. Unless stated otherwise, we use the following type conventions for variables

$$i, j, k, l, m, n \colon \mathbf{N},\ s, t \colon \rho^*, \text{ and } \alpha, \beta \colon \rho^\omega,$$

where ρ is an arbitrary type.

To enhance readability, we often insert parentheses when writing applications, e.g., we write $M(x, y, z)$ for $Mxyz$.

A predicate $P(\vec{x}^{\vec{\rho}})$ is called *(higher type) primitive recursive* if there is a closed term $\chi_P \colon \vec{\rho} \to \mathbf{N}$ such that $\mathsf{HA}^\omega \vdash P(\vec{x}) \leftrightarrow \chi_P(\vec{x}) =_{\mathbf{N}} 0$.

In general, we won't be very formal when defining new terms or predicates of HA^ω. However, all definitions are such that they can be easily translated into definitions using the structural recursion operators provided by our theory. We presuppose all standard definitions and notations for operations on natural numbers (e.g., addition $+$ and cut-off subtraction $\dot{-}$) and of primitive recursive predicates between them; for $n \in \mathbb{N}$ we sometimes just write n instead of the numeral $S^n 0$. Case distinction is written as **if** n **then** x^ρ **else** y^ρ and satisfies **if** 0 **then** x **else** $y =_\rho x$ and **if** (Sn) **then** x **else** $y =_\rho y$; if P is a primitive recursive predicate we write **if** $P(\vec{x})$ **then** x **else** y for **if** $\chi_P(\vec{x})$ **then** x **else** y. Often such case distinctions will be displayed using brackets (see below). Recall, that pairing is denoted by $\langle \cdot, \cdot \rangle$, left projection by π_0, and right projection by π_1; adding an element

x^ρ to a list s^{ρ^*} is denoted by $s * x$ and $\langle\rangle$ is the empty list. We use the notation $\langle x_0^\rho, \dots, x_{n-1}^\rho\rangle$ to denote the list of type ρ^* with elements $x_0, \dots, x_{n-1}$. The length of the list s is denoted by $|s|$, so if $s = \langle x_0, \dots, x_{n-1}\rangle$, then $|s| = n$.

The *canonical embedding* $\mathbf{emb}_\rho \colon \mathbf{N} \to \rho$ of $\mathbf{N}$ into ρ is defined by induction on the type

$$\mathbf{emb}_\mathbf{N}(n) := n, \qquad \mathbf{emb}_{\rho\times\sigma}(n) := \langle \mathbf{emb}_\rho(n), \mathbf{emb}_\sigma(n)\rangle,$$
$$\mathbf{emb}_{\rho^*}(n) := \langle \mathbf{emb}_\rho(n)\rangle, \qquad \mathbf{emb}_{\rho\to\sigma}(n) := \lambda x^\rho.\, \mathbf{emb}_\sigma(n).$$

Its inverse operation $\psi_\rho \colon \rho \to \mathbf{N}$ is defined by induction on the type:[1]

$$\psi_\mathbf{N}(n) := n \quad \psi_{\rho\to\sigma}(f) := \psi_\sigma(f(\mathbf{emb}_\rho\, 0)) \quad \psi_{\rho\times\sigma}(x) := \psi_\rho(\pi_0(x))$$
$$\psi_{\rho^*}(s) := \mathbf{if}\ s = \langle\rangle\ \mathbf{then}\ 2^\mathbf{N}\ \mathbf{else}\ \psi_\rho(s_0)$$

Here s_0 is the first element of s. By induction on ρ one immediately verifies that (provably in HA^ω)

$$\psi_\rho(\mathbf{emb}_\rho\, n) = n.$$

For $n \colon \mathbf{N}$ we also write n^ρ for $\mathbf{emb}_\rho(n)$ and sometimes just n if ρ is clear from the context and n is a numeral.

For a list s, s_n denotes the $(n{+}1)$-th element of s if $n < |s|$, and otherwise $s_n = 0^\rho$. In particular, $\langle x_0, \dots, x_{n-1}\rangle_m = x_m$ for $m < n$, and for each list s, $s = \langle s_0, \dots, s_{|s|-1}\rangle$.

We need the following notations concerning lists and sequences:

$$\begin{aligned}
\bar\alpha n &= \langle \alpha(0), \dots, \alpha(n-1)\rangle,\\
(s \mathbin{@} \alpha)(n) &= \mathbf{if}\ n < |s|\ \mathbf{then}\ s_n\ \mathbf{else}\ \alpha(n),\\
(s * \alpha)(n) &= \mathbf{if}\ n < |s|\ \mathbf{then}\ s_n\ \mathbf{else}\ \alpha(n \dot- |s|),\\
s * t &= \text{concatenation of } s \text{ and } t,\\
\alpha \in s &:\leftrightarrow \bar\alpha|s| = s,\\
s \in t &:\leftrightarrow |s| \le |t| \wedge \forall i < |s| (s_i = t_i),\\
\hat s &= s \mathbin{@} 0^{\rho^\omega},\\
(\overline{\alpha, n}) &= \bar\alpha n \mathbin{@} 0^{\rho^\omega}.
\end{aligned}$$

Note that the overloading of the operator "$*$" does not pose a problem because the type of the operands always determine which "$*$" is meant.

Often we write $\mathbf{if}\ n\ \mathbf{then}\ x^\rho$ instead of the longer $\mathbf{if}\ n\ \mathbf{then}\ x^\rho\ \mathbf{else}\ 0^\rho$.

1. Bar Recursion and Variants

In his last article [**42**], Spector extended Gödel's Dialectica interpretation of arithmetic to classical analysis. This was achieved by adjoining a new scheme to Gödel's T, called bar recursion.[2] His main achievement is

---

[1] The magic number 2 appearing in the definition is inessential and only used for a technical argument in the proof of Lemma 5.4.

[2] This, however, was already suggested by Gödel in [**18**, p. 286] where he writes: "Es ist klar, dass man, von demselben Grundgedanken ausgehend, auch viel stärkere Systeme als T konstruieren kann, zum Beispiel durch Zulassung transfiniter Typen oder der von Brouwer für den Beweis des 'Fan-Theorems' benutzten Schlussweise." (It is clear that, starting from the same basic idea, one can construct much stronger systems than T, for

the proof-theoretic reduction via the Dialectica interpretation of classical analysis to a *quantifier-free* system build from T extended with constants for bar recursion. For more details we refer the reader to Spector's article.

Later we will see that bar recursion may be regarded as recursion over well-founded trees given by higher type functionals.

*Bar recursion*is given by the following equation:

$$\mathrm{SBR}_{\rho,\tau}(\Phi) \qquad \Phi YGHs =_\tau \begin{cases} G(s) & \text{if } Y(\hat{s}) < |s|, \\ H(s, \lambda x^\rho.\Phi YGH(s * x)) & \text{otherwise.} \end{cases}$$

Here Y has type $\rho^\omega \to \mathbf{N}$, G has type $\rho^* \to \tau$, and H has type $\rho^* \to (\rho \to \tau) \to \tau$. Formally $\mathrm{SBR}_{\rho,\tau}(\Phi)$ stands for the universally quantified predicate (with distinguished variable Φ)

$$\forall Y, G, H, s.\Phi YGHs =_\tau \begin{cases} G(s) & \text{if } Y(\hat{s}) < |s|, \\ H(s, \lambda x^\rho.\Phi YGH(s * x)) & \text{otherwise.} \end{cases}$$

We write $\mathrm{SBR}_{\rho,\tau}$ for $\exists\Phi\,\mathrm{SBR}_{\rho,\tau}(\Phi)$ and SBR for the union of all $\mathrm{SBR}_{\rho,\tau}$, $\rho, \tau \in \mathsf{Ty}$. The same conventions apply in future definitions.

Our definition follows the one given by Howard in [**21**], where the only difference to Spector's definition in [**42**] is that instead of using a type for finite lists, Spector used a sequence $C\colon \mathbf{N} \to \rho$ and a natural number x to encode the list $\langle C0, \ldots, C(x-1)\rangle$. It was shown by Bezem in [**9**, p. 155] that both variants are equivalent.

We now give an informal explanation of how one can view bar recursion as recursion over well-founded trees. In order to give this intuition, we first need some notions.

A set B of finite sequences (of a fixed set M) *bars* a node s, if

$$\forall \alpha \in s \exists n\, \bar{\alpha}n \in B.$$

If B bars $\langle\rangle$, then we simply call B a *bar*. A set T of finite sequences is called a *tree* if T is closed under initial segments. For a node $s \in T$ define its *immediate successors* in T by $\mathrm{succ}_T(s) := \{s * x \mid s * x \in T \wedge x \in M\}$. A node $s \in T$ is a *leaf* if it possesses no immediate successor in T. A tree T is *well-founded* if it has no infinite path, i.e.,

$$\forall \alpha \exists n\, \bar{\alpha}n \notin T.$$

Now assume that B is a bar and that B is *decidable*, i.e.,

$$\forall s (s \in B \vee s \notin B).$$

We define

$$T_B := \{s \mid \forall t \in s\; t \notin B\}.$$

Then T_B is a well-founded tree and we can specify a function $F\colon T_B \cup B \to X$ by *bar recursion on* T_B by specifying F on B by G and on T_B by H such

example by admitting transfinite types or the rule of inference which Brouwer uses to prove the "Fan theorem". [**19**]) that[3]

$$F(s) = \begin{cases} G(s) & \text{if } s \in B, \\ H(s, F \restriction \mathrm{succ}_T(s)) & \text{otherwise.} \end{cases}$$

In particular, F is defined on $\langle\rangle$. Now Spector's bar recursion in the point $\langle\rangle$ (given functionals Y, G and H) is given by bar recursion on the tree T_{B_Y} where

$$B_Y := \{s \mid Y(\hat{s}) < |s|\}.$$

Of course, Y must be such that B_Y is indeed a bar. Moreover, bar recursion is uniform in the arguments, in particular in B_Y. This indicates that finding models for bar recursion is non-trivial. In fact, the full set-theoretical structure[4] is not a model of bar recursion.[5]

Our next goal is to prove that SBR is indeed valid in the model of Kleene-Kreisel functionals $\mathbb{G}$. For this we first need the principle of *bar induction for decidable bars* BI_D given by:

$$\begin{aligned} &\forall s(P(s) \vee \neg P(s)) \wedge \\ &\forall \alpha \exists n P(\bar{\alpha} n) \wedge \\ &\forall s(P(s) \to Q(s)) \wedge \\ &\forall s(\forall x^\rho Q(s * x) \to Q(s)) \ \to \\ &Q(\langle\rangle). \end{aligned}$$

Here P and Q range over $\mathsf{HA}^\omega[\mathcal{X}]$-predicates of arity (ρ^*).

Note that BI_D follows from DC with classical logic: assuming $\neg Q(\langle\rangle)$ we can construct with DC an infinite sequence α using $\neg Q(s) \to \exists x \neg Q(s * x)$ such that $\neg Q(\bar{\alpha} n)$ for all n. Therefore $\neg P(\bar{\alpha} n)$ for all n, i.e., P is not a bar.[6]

It is convenient to consider the following modification BI'_D of BI_D: BI'_D is like BI_D, but instead of the premise $\forall \alpha \exists n P(\bar{\alpha} n)$ we take $\forall \alpha \forall k \exists n \geq k\, P(\bar{\alpha} n)$ (we say that P is an *infinite bar*), and instead of the conclusion $Q(\langle\rangle)$ we take $\forall s Q(s)$.

Lemma 1.1. $\mathsf{HA}^\omega[\mathcal{X}] + \mathrm{BI}_D \vdash \mathrm{BI}'_D$.

Proof. Let predicates $P(s)$ and $Q(s)$ as above be given and assume the premises of BI'_D. For a fixed s we must prove $Q(s)$. Define

$$Q_s(t) :\leftrightarrow Q(s * t), \text{ and } P_s(t) :\leftrightarrow P(s * t).$$

[3]In a set-theoretical framework the existence of F is guaranteed by the recursion theorem for well-founded relations because the (proper) initial segment relation is well-founded since T_B is so.

[4]In the set-theoretical model $\mathcal{S}$ one has $\mathcal{S}_{\mathbf{N}} = \mathbb{N}$ and $\mathcal{S}_{\rho \to \sigma}$ is defined to be *all* functions from $\mathcal{S}_\rho$ to $\mathcal{S}_\sigma$; product and list types are interpreted as products and finite sequences respectively.

[5]Consider, e.g., the functional $Y\colon (\mathbb{N} \to \mathbb{N}) \to \mathbb{N}$ defined by $Y(\alpha) = n$ if α has n roots, and $= 0$ if α has infinitely many roots. Then with $H(s, F) := 1 + F(1)$ and G arbitrary, one easily sees that, if there would be a function Ψ satisfying the equation for bar recursion in the set-theoretical model, then $\Psi Y G H \langle\rangle = \Psi Y G H (1^n) + n \geq n$ for each n, which is impossible.

[6]This argument even shows that the stronger principle of *classical* bar induction BI, given by (B arbitrary)

$$\forall \alpha \exists n B(\bar{\alpha} n) \wedge \forall s(\forall x B(s * x) \to B(s)) \to B(\langle\rangle),$$

is derivable using classical logic and dependent choice.

We have to show $Q_s(\langle\rangle)$. This is proved by BI_D for the predicates Q_s and P_s. It remains to verify the premises of BI_D. We only show that P_s is a bar, i.e., $\forall\alpha\exists n P_s(\bar{\alpha}n)$. The rest follows easily from the respective assumptions on P and Q. Let α be given. Since P is an infinite bar we have

$$\forall\beta\forall k\exists n \geq k\, P(\bar{\beta}n),$$

in particular, for $s * \alpha$ and $|s|$ there exists $n \geq |s|$ such that $P((\overline{s * \alpha})n)$, and hence $P_s(\bar{\alpha}(n - |s|))$. □

REMARK 1.2. Another variant of bar induction for decidable bars BI_D considered in the literature is *bar induction for monotone bars* BI_M, where the decidability of P is replaced with the monotonicity condition $P(s) \to P(s * x)$ for all s and x^ρ. (Cf. [**22**] for a comparison and further discussion.)

The principle BI'_D is similar to Kohlenbach's $(\mathrm{BI})^\rho_*$ [**24**, p. 7], where the only difference is that he uses the premise $\forall\alpha\exists k\forall n \geq kP(\bar{\alpha}n)$ instead of our $\forall\alpha\forall k\exists n \geq kP(\bar{\alpha}n)$. For our treatment, it is not essential which of the two variants one chooses.

NOTATION. To not complicate our notation too much, we often identify terms with their denotation, e.g., for the term $\lambda x^{\rho^*}.\hat{x} \colon \rho^* \to \rho^\omega$ and an ideal s of type ρ^*, we write $\hat{s}$ for the ideal $[\![\lambda x^{\rho^*}.\hat{x}]\!](s)$.

THEOREM 1.3. *Spector's bar recursion exists in the model* $\mathbf{G}$*, more precisely* $\mathbf{G} \models \mathrm{SBR}$*. Moreover, with the transfer principle* $\mathbb{G} \models \mathrm{SBR}$*.*

PROOF. Let D be a defined constant with computation rule (additional to the computation rules of system T):

$$DYGHs \rhd \textbf{if}\ Y(\hat{s}) < |s|\ \textbf{then}\ G(s)\ \textbf{else}\ H(s, \lambda x^\rho.DYGH(s * x)). \tag{9}$$

By the results of Section 4 in Chapter 2, it suffices to show that $\Phi := [\![D]\!] \in |\mathbf{C}|$ is total.[7] So fix total Y, G, and H. We have to show that $\Psi(s) := \Phi YGHs$ is total for all total s.

Let Q be a predicate variable of arity (ρ^*) and define the $\{Q\}$-assignment Ξ by

$$\Xi(Q) = \{s \in \mathbf{G}_{\rho^*} \mid \Psi(s) \text{ is total}\}.$$

Then, as DC implies BI'_D classically, we have $\mathbf{G}, \Xi \models \mathrm{BI}'_D$. We now prove $\mathbf{G}, \Xi \models \forall s Q(s)$ by BI'_D with

$$P(s) :\leftrightarrow Y(\hat{s}) < |s|.$$

That P is decidable in $\mathbf{G}$ is trivial. We now verify the remaining premises of BI'_D (where we argue in $\mathbf{G}, \Xi$).

(i) $\forall\alpha\forall k\exists n \geq k\, Y(\overline{\alpha, n}) < n$. Let total α and k be given. By continuity of Y there exists m such that $Y(\beta) = Y(\alpha)$ for all $\beta \in \bar{\alpha}m$. Let $n := \max\{k, m, Y(\alpha)\} + 1$. Then $(\overline{\alpha, n}) \in \bar{\alpha}m$ and therefore $Y(\overline{\alpha, n}) = Y(\alpha) < n$.

(ii) $\forall s(P(s) \to Q(s))$. If $P(s)$ for a total s, then $\Psi(s) = G(s)$ is total because G and s are.

[7] Alternatively, one could also use fixed-point operators to define Φ.

(iii) $\forall s(\forall x Q(s*x) \to Q(s))$. Let s be total with $Q(s*x)$ for all total x. We have to show that $\Psi(s)$ is total. We may assume $\neg P(s)$, i.e., $Y(\hat{s}) \geq |s|$. By assumption, we obtain that $\Psi(s*x)$ is total for all total x, and hence

$$[\![\lambda y, g, h, t\lambda x^\rho.Dygh(t*x)]\!](Y, G, H, s) \text{ is total.}$$

Therefore $\Psi(s) = H\big(s, [\![\lambda y, g, h, t\lambda x^\rho.Dygh(t*x)]\!](Y, G, H, s)\big)$ is total, since H and s are total. □

Remark 1.4 (Strong normalization; adequacy). Let $\longrightarrow$ denote the reduction relation $\longrightarrow_{\beta\mathcal{P}}$, where $\mathcal{P}$ is the system of computation rules of Gödel's T together with defined constants D and the computation rules as in (9) of the proof of the last theorem. Then the reduction relation $\longrightarrow$ is *not* strongly normalizing because, for $D' := DYGH$ with D a bar recursor, we have

$$D's \longrightarrow \textbf{if } Y(\hat{s}) < |s| \textbf{ then } G(s) \textbf{ else } H(s, \lambda x^\rho.D'(s*x)).$$

Reducing the innermost D' again and again, leads to an infinite reduction sequence.

However, this defect can be repaired if one forces the test $Y(\hat{s}) < |s|$ to take place before reducing the bar recursor: Let B be a defined constant with the following computation rules

$$\begin{aligned} BYGHs\,0 &\rhd G(s) \colon \tau, \\ BYGHs\,(Sn) &\rhd H(s, \lambda x^\rho.BYGH(s*x)(Y(\widehat{s*x}) \dot{-} |s|)). \end{aligned}$$

Then the denotational semantics of

$$\lambda YGHs.BYGHs((Y(\hat{s})+1) \dot{-} |s|)$$

is total as well, satisfies the equation of SBR, and the resulting reduction relation is strongly normalizing. This was first proved in a combinatorial formulation by Vogel in [**49**] (where the above trick was also introduced) extending the weak normalization proof of Tait [**44**]; a proof for the formulation in λ-calculus can be found in [**5**].

Although $\longrightarrow$ is not strongly normalizing, it can be used to effectively compute the normal form of a closed term of type $\mathbf{N}$. This is guaranteed by the Adequacy Theorem (cf. Section 5 in Chapter 2) which even gives us a deterministic reduction strategy (by our operational semantics). Note that this can as well be done for all recursion schemes below for which we prove totality – in particular, our realizers in the next chapter.

In his Ph.D. thesis [**24**], Kohlenbach introduced another variant of Spector's bar recursion. In his version, the stopping condition $Y(\hat{s}) < |s|$ is replaced with $Y(s \mathbin{@} 0) = Y(s \mathbin{@} 1)$. It turns out, that this variant is strictly stronger than Spector's original formulation (cf. [**24**, p. 57 ff.]). Altogether, the equation for *Kohlenbach's variant of bar recursion* is

$$\mathrm{KBR}_{\rho,\tau}(\Phi) \qquad \Phi YGHs =_\tau \begin{cases} G(s) & \text{if } Y(s \mathbin{@} 0) =_{\mathbf{N}} Y(s \mathbin{@} 1), \\ H(s, \lambda x^\rho.\Phi YGH(s*x)) & \text{otherwise.} \end{cases}$$

Theorem 1.5. $\mathbf{G} \models \mathrm{KBR}$ *and* $\mathbb{G} \models \mathrm{KBR}$.

PROOF. Analogously to the proof of Theorem 1.3 but with

$$P(s) :\leftrightarrow Y(s \mathbin{@} 0) = Y(s \mathbin{@} 1).$$

That P is an infinite bar follows immediately from the continuity of Y. □

2. Modified Bar Recursion

2.1. The Berardi-Bezem-Coquand Functional. The Berardi-Bezem-Coquand (BBC) functional was introduced in [**2**] in order to give a (new) computational interpretation of the classical axiom of choice. For this they gave a realizer of the negative translated axiom of choice w.r.t. a non-standard version of modified realizability and A-translation. Their setting also employs fixed-point operators (in order to define their functional) and infinite terms. The latter were only needed in the correctness proof of their realizer.

In order to define the BBC functional, we first need an auxiliary functional. For $s\colon (\mathbf{N}\times\rho)^*$ (which we can view as an approximation of an infinite sequence) and $\alpha\colon \rho^\omega$ we define $s\{\alpha\}\colon \rho^\omega$ by

$$(\langle\rangle\{\alpha\})n := \alpha n,$$
$$((s * \langle m, x\rangle)\{\alpha\})n := \begin{cases} x & \text{if } n = m, \\ (s\{\alpha\})n & \text{otherwise.} \end{cases}$$

Now the *BBC functional* is given by the equation

$$\mathrm{BBC}_\rho(\Psi) \qquad \Psi Y H s =_{\mathbf{N}} Y\big(s\{\lambda n.H(s, n, \lambda x^\rho.\Psi Y H(s * \langle n, x\rangle))\}\big),$$

where $s\colon (\mathbf{N} \times \rho)^*$ and all other types can be inferred from the context.

We also consider another variant of the BBC functional, which we call *weak BBC functional*, given by

$$\mathrm{wBBC}_\rho(\Psi) \qquad \Psi Y H s =_{\mathbf{N}} Y\big(s\{\lambda n.H(n, \lambda x^\rho.\Psi Y H(s * \langle n, x\rangle))\}\big).$$

The weak BBC functional suffices for the realizer given in [**2**], but it is an open problem whether it is equivalent to BBC.

2.2. Modified Bar Recursion. Inspired by the BBC functional, Berger and Oliva introduced a variation of bar recursion in [**6**] which they named *modified bar recursion*. The main difference to the approach of Berardi, Bezem, and Coquand is that their functional provides a realizer of the negative translated axiom of dependent choice w.r.t. standard definitions of modified realizability and A-translation, and allows an easier correctness proof. In Chapter 4 this will be done in detail.

Modified bar recursion is given by the equation

$$\mathrm{MBR}_\rho(\Psi) \qquad \Psi Y H s =_{\mathbf{N}} Y\big(s \mathbin{@} H(s, \lambda x^\rho.\Psi Y H(s * x))\big).$$

Here Y has type $\rho^\omega \to \mathbf{N}$ and H has type $\rho^* \to (\rho \to \mathbf{N}) \to \rho^\omega$.

Another variant of this is *weak modified bar recursion* given by

$$\mathrm{wMBR}_\rho(\Psi) \qquad \Psi Y H s =_{\mathbf{N}} Y\big(s \mathbin{@} \lambda n.H(s, \lambda x^\rho.\Psi Y H(s * x))\big).$$

Note that here H has type $\rho^* \to (\rho \to \mathbf{N}) \to \rho$.

Remark 2.1. It is essential that the equation for MBR and wMBR is at a type of level zero. Suppose there is a functional Ψ satisfying the equation for wMBR at, say, type $\mathbf{N} \to \mathbf{N}$, i.e.,

$$\Psi Y H s =_{\mathbf{N}\to\mathbf{N}} Y\big(s \mathbin{@} \lambda n.H(s, \lambda x^\rho.\Psi Y H(s * x))\big).$$

Then for $Y(\alpha, m) := \alpha(m) + 1$ and $H(s, F) := F(0, |s| + 1)$ we obtain

$$\Psi Y H s m =_{\mathbf{N}} \big(s \mathbin{@} \lambda n.\Psi Y H(s * 0, |s| + 1))\big)(m) + 1,$$

and thus, by induction on m,

$$\Psi Y H\langle\rangle 0 = \Psi Y H \underbrace{\langle 0, \ldots, 0\rangle}_{m}(m) + m$$

which is inconsistent with HA^ω.

Our next aim is the totality of the modified bar recursor. This was shown in [**6**] although the argument there was only sketched. Here we give a proof adapted to our setting. For this we need some preparatory observations concerning continuity properties of the partial continuous functionals. Recall the definition of $s \mathbin{@} \alpha$ as a term of T:

$$s \mathbin{@} \alpha = \lambda n.\textbf{if } n < |s| \textbf{ then } s_n \textbf{ else } \alpha(n).$$

When dealing with total objects only, it is not essential how $<$ and if-terms are implemented as HA^ω-terms. When dealing with partial objects, however, different implementations result in different behavior on non-total objects. We assume the following properties of the denotation of $<$ and if-terms (which we also write as $<$ and **if** $\cdot$ **then** $\cdot$ **else** $\cdot$):

$$Sx < Sy \text{ iff } x < y, 0 < Sy, \neg Sx < 0, \emptyset < x \text{ is undefined}$$
$$\text{(i.e., the characteristic function is } \emptyset\text{), and } \textbf{if } \emptyset \textbf{ then } z_1 \textbf{ else } z_2 = \emptyset,$$

for $x, y \in |\mathbf{C}_\mathbf{N}|$ and $z_1, z_2 \in |\mathbf{C}_\rho|$.

Let $\bot := \emptyset$, $\alpha \in \mathbf{G}_{\rho^\omega}$, $n \in \mathbb{N}$, and $x \in |\mathbf{C}_\mathbf{N}|$ with x not total, i.e., $x \notin \mathbf{G}_\mathbf{N}$. We now claim that $(\bar{\alpha}n \mathbin{@} \bot)(x) = \bot$. In case x is of the form $S^m y$ for some $y \in |\mathbf{C}_\mathbf{N}|$ and $m > n$, then, using the assumptions on the implementation from above, we get $\neg(S^m y < n)$, and hence $(\bar{\alpha}n \mathbin{@} \bot)(x) = \bot(x) = \bot$. Otherwise, we can write x as $S^m\bot$ with $m \le n$. Then $x < n$ iff $\bot < n - m$, which is undefined. Hence, by the assumption on if-terms, we conclude $(\bar{\alpha}n \mathbin{@} \bot)(x) = \bot$. This proves the claim.

From the claim above we conclude for $n, m \in \mathbb{N}$ with $n \le m$ that[8]

$$\bar{\alpha}n \mathbin{@} \bot \subseteq \bar{\alpha}m \mathbin{@} \bot,$$

and hence the set

$$\check{\alpha} := \{\bar{\alpha}n \mathbin{@} \bot \mid n \in \mathbb{N}\} \subseteq |\mathbf{C}_{\rho^\omega}|,$$

is directed. Moreover, the claim also yields $\bigcup \check{\alpha} = \alpha_\bot$, where $\alpha_\bot$ is defined as in Section 7.2 of Chapter 2.

As a consequence of our considerations, we obtain:

[8]Here we need extensionality for partial continuous functionals, i.e., $f = g \in |\mathbf{C}_{\rho\to\sigma}|$ whenever $f(x) = g(x)$ for all $x \in |\mathbf{C}_\rho|$. It is not hard to see that even equality on ideals of the form $\overline{U}$ for $U \in \mathrm{Con}_\rho$ suffices.

LEMMA 2.2. *Let $F \in |\mathbf{C}_{\rho^\omega \to \mathbf{N}}|$, $\alpha \in \mathbf{G}_{\rho^\omega}$, and $n \in \mathbb{N}$. Then:*

$$F(\alpha_\perp) = n \leftrightarrow \exists m \in \mathbb{N}. F(\bar{\alpha}m \mathbin{@} \perp) = n.$$

PROOF. Suppose $F(\bar{\alpha}m \mathbin{@} \perp) = n$ for some $m \in \mathbb{N}$. Because F is monotone and $\bar{\alpha}m@\perp \subseteq \alpha_\perp$ we get $n = F(\bar{\alpha}m@\perp) \subseteq F(\alpha_\perp)$. Since n is maximal, we conclude $F(\alpha_\perp) = n$.

Conversely, assume $F(\alpha_\perp) = n$. Because $\check{\alpha}$ is directed, $\bigcup \check{\alpha} = \alpha_\perp$, and F is continuous, we obtain:

$$n = F(\alpha_\perp) = F(\bigcup \check{\alpha}) = \bigcup_{m \in \mathbb{N}} F(\bar{\alpha}m \mathbin{@} \perp).$$

Hence $S^n 0 \in F(\bar{\alpha}m \mathbin{@} \perp)$ for some $m \in \mathbb{N}$, i.e., $n = F(\bar{\alpha}m \mathbin{@} \perp)$ for some $m \in \mathbb{N}$. □

THEOREM 2.3. $\mathbf{G} \models \mathrm{MBR}$ *and* $\mathbb{G} \models \mathrm{MBR}$.

PROOF. Let D be a defined constant with computation rule (additional to the computation rules of system T)

$$DYHs \rhd Y\big(s \mathbin{@} H(s, \lambda x^\rho.DYH(s * x))\big),$$

where Y has type $\rho^\omega \to \mathbf{N}$.

Analogously to Theorem 1.3, it suffices to show that $\Psi := [\![D]\!] \in |\mathbf{C}|$ is total. Fix total Y and H. We have to show that $\Psi'(s) := \Psi Y H s \in \mathbb{N}$ for all total s.

Let P and Q be predicate variables of arity (ρ^*). We define the $\{P, Q\}$-assignment Ξ by

$$\Xi(P) := \{s \in \mathbf{G}_{\rho^*} \mid Y(s \mathbin{@} \perp) \text{ is total}\},$$
$$\Xi(Q) := \{s \in \mathbf{G}_{\rho^*} \mid \Psi'(s) \text{ is total}\}.$$

The claim follows from $\mathbf{G}, \Xi \models \forall s Q(s)$, which we prove using BI'_D for P and Q (in $\mathbf{G}$). That P is decidable is trivial. We now verify the remaining premises of BI'_D (where we argue informally in $\mathbf{G}, \Xi$).

(i) $\forall \alpha \forall k \exists n \geq k\, Y(\bar{\alpha}n \mathbin{@} \perp)$ is total. Let α be total. Since Y is monotone it suffices to prove $Y(\bar{\alpha}n \mathbin{@} \perp)$ is total for some n. Because α is total, so is $\alpha_\perp$ and $\alpha \sim \alpha_\perp$, and hence by the totality of Y, $Y(\alpha) = Y(\alpha_\perp)$ is total. Thus the claim follows from Lemma 2.2.

(ii) $\forall s(P(s) \to Q(s))$. Fix total s with $P(s)$, i.e., $Y(s \mathbin{@} \perp)$ is total. By the monotonicity of Y and the defining equation for D we conclude $Y(s \mathbin{@} \perp) \subseteq \Psi'(s)$, and hence $\Psi'(s)$ is total, i.e., $Q(s)$.

(iii) $\forall s(\forall x Q(s * x) \to Q(s))$. Let s be total such that $\Psi'(s * x)$ is total for all total x's. It follows that

$$[\![\lambda y, h, t, \lambda x^\rho.Dyh(t * x)]\!](Y, H, s) \text{ is total.}$$

Therefore $\Psi'(s) = Y\big(s \mathbin{@} H(s, [\![\lambda y, h, t, \lambda x^\rho.Dyh(t * x)]\!](Y, H, s))\big)$ is total because Y, H, and s are total. □

3. Open Recursion and Variants

Open recursion was first introduced by Berger in [**4**] and is the computational counterpart of (a fragment of) open induction. This principle was introduced by Raoult [**34**] in a classical context and later analyzed in a constructive setting by Coquand in [**10**] and others. The fragment of the principle studied by Raoult can also be seen as a classical reformulation of the minimal-bad-sequence argument of Nash-Williams [**30**] (cf. the proof of 3.6). Roughly speaking, open induction is induction over sequences ordered lexicographically but restricted to open predicates (w.r.t. to the topology indicated in Footnote 4 of Chapter 2). This restriction is necessary because, in general, the lexicographical ordering on sequences is not well-founded. In this section, we also consider update recursion and induction from [**4**] and introduce extensions thereof, which we call extended update recursion and induction.

It is illustrative to talk about open and transfinite induction first. In order to do so, we need some additional notions.

Definition 3.1. Let $\prec$ be a binary relation on ρ and let P be a $\mathsf{HA}^\omega[\mathcal{X}]$-predicate of arity (ρ).

(1) The $\prec$-*progressiveness* $\mathrm{Prog}_\prec(P)$ of P is defined as:

$$\mathrm{Prog}_\prec(P) :\leftrightarrow \forall x(\forall y \prec x P(y) \to P(x)).$$

(2) Let $\prec$ be primitive recursive. *Transfinite induction* for P is given by

$$\mathrm{TI}_{\prec,P} \quad \mathrm{Prog}_\prec(P) \to \forall x P(x).$$

By $\mathrm{TI}_\prec$ we denote the scheme of transfinite induction (i.e., all $\mathrm{TI}_{\prec,P}$, P an arbitrary $\mathsf{HA}^\omega[\mathcal{X}]$-predicate of arity (ρ)). We also say that $\mathrm{TI}_\prec$ expresses the well-foundedness of $\prec$.

(3) We define the *lexicographical extension* $\prec_{\mathsf{lex}}$ of $\prec$ to ρ^ω by

$$\alpha \prec_{\mathsf{lex}} \beta :\leftrightarrow \exists n.\bar{\alpha}n = \bar{\beta}n \wedge \alpha n \prec \beta n.$$

Definition 3.2. (1) A $\mathsf{HA}^\omega[\mathcal{X}]$-formula B is called a Σ-*formula* if all of the following hold:

(i) B is $\to\forall$-free.
(ii) If P is a predicate symbol of arity $(\rho_1, \ldots, \rho_n)$ occurring in B, then $\mathrm{lev}(\rho_i) = 0$ for all $i = 1, \ldots, n$.
(iii) The type of any occurrence of a variable bound by an existential quantifier has level 0.

(2) For a predicate C of arity (ρ^*) and a quantifier $\mathsf{Q} \in \{\forall, \exists\}$ we define the predicate $C^{\mathsf{Q}}(\alpha) := \mathsf{Q}nC(\bar{\alpha}n)$ of arity (ρ^ω).

(3) Let U be a $\mathsf{HA}^\omega[\mathcal{X}]$-predicate. U is called *open* (w.r.t. ρ) if U has arity (ρ^ω) and there are predicates B and C of arity (ρ^*) such that B is a Σ-formula and

$$U(\alpha) = C^\forall(\alpha) \to B^\exists(\alpha).$$

(4) *Open induction* is given by:

$$\mathrm{OI}_{\prec,U} \qquad \mathrm{Prog}_{\prec_{\mathsf{lex}}}(U) \to \forall \alpha U(\alpha)$$

i.e.,

$$\forall\alpha\,(\forall\beta \prec_{\mathsf{lex}} \alpha\, U(\beta) \to U(\alpha)) \to \forall\alpha U(\alpha),$$

where U ranges over open $\mathsf{HA}^\omega[\mathcal{X}]$-predicates, ρ over types, and $\prec$ ranges over primitive recursive binary relations of arity (ρ,ρ) with $\mathsf{HA}^\omega[\mathcal{X}] \vdash \mathrm{TI}_\prec$.

REMARK 3.3. (1) Unless $\prec$ is trivial, $\prec_{\mathsf{lex}}$ is *not* well-founded. For instance, if $0 \prec 1$, then

$$1000\cdots \succ_{\mathsf{lex}} 0100\cdots \succ_{\mathsf{lex}} 0010\cdots \succ_{\mathsf{lex}} \cdots$$

is an infinite descending sequence in the Cantor space.

(2) Classically, any predicate of the form $C^\forall(\alpha) \to B^\exists(\alpha)$ with C and B arbitrary, is equivalent to $C^\forall(\alpha) \wedge \neg B^\forall(\alpha) \to \bot$ which is open.

Moreover, with classical logic, for any open predicate U there is a predicate D such that (provably in $\mathsf{PA}^\omega[\mathcal{X}]$):

$$\forall\alpha.\, U(\alpha) \leftrightarrow \exists n D(\bar{\alpha}n).$$

Hence, classically, open predicates are open w.r.t. the usual topology on sequences (cf. Footnote 4 in Chapter 2).

LEMMA 3.4. *Open predicates are extensional, i.e., if U is an open predicate, then:*

$$\mathsf{HA}^\omega[\mathcal{X}] \vdash \forall\alpha,\beta\,(\forall n\, \alpha n =_\rho \beta n \to U(\alpha) \to U(\beta))\,.$$

PROOF. Let U be open, i.e., $U(\alpha) = C^\forall(\alpha) \to B^\exists(\alpha)$ for C arbitrary and B a Σ-formula. Let $\alpha,\beta\colon \rho^\omega$ and suppose

$$\forall n\, \alpha n =_\rho \beta n, \tag{10}$$

$U(\alpha)$, and $C^\forall(\beta)$. We have to show $B^\exists(\beta)$, i.e., $B(\bar{\beta}n)$ for some n. Using (10) one easily proves by induction on n, that

$$\forall n\, \bar{\alpha}n =_{\rho^*} \bar{\beta}n$$

and hence $C^\forall(\beta)$, i.e., $\forall n\, C(\bar{\beta}n)$ implies $C^\forall(\alpha)$ using compatibility for type ρ^*. Together with $U(\alpha)$ we obtain $B^\exists(\alpha)$, i.e., $B(\bar{\alpha}n)$ for some n and hence again with compatibility $B(\bar{\beta}n)$. □

It was observed by Troelstra in [**47**, p. 226–228] that transfinite induction $\mathrm{TI}_\prec$ (for $\prec$ primitive recursive) is realized (in the sense of the modified realizability interpretation; cf. Chapter 4) by *transfinite recursion*

$$\mathrm{TR}_\prec(\mathcal{R}^\prec) \qquad \mathcal{R}^\prec F x = F x (\lambda y^\rho \textbf{if } x \prec y \textbf{ then } \mathcal{R}^\prec F y).$$

Note that here $\prec$ is decidable, but in general $\prec_{\mathsf{lex}}$ is not. However, we can reformulate open induction as

$$\forall\alpha\,\big(\forall n, x^\rho, \gamma^{\rho^\omega} (x \prec \alpha n \to U(\bar{\alpha}n * x \mathbin{@} \gamma)) \to U(\alpha)\big) \to \forall\alpha U(\alpha). \tag{11}$$

This scheme is (provably in $\mathsf{HA}^\omega[\mathcal{X}]$) equivalent to the formulation above using

$$\forall n, x^\rho, \gamma\big(x \prec \alpha n \to U(\bar{\alpha}n * x \mathbin{@} \gamma)\big) \leftrightarrow \forall\beta \prec_{\mathsf{lex}} \alpha\, U(\beta),$$

which follows from the extensionality of open predicates.

This reformulation of open induction suggests the following recursion scheme, which we call *open recursion*:

$$\mathrm{OR}_{\prec,\rho}(\mathcal{R}^{\mathsf{o}}_{\prec}) \quad \mathcal{R}^{\mathsf{o}}_{\prec} F\alpha =_{\mathbf{N}} F\alpha\big(\lambda n, x^{\rho}, \beta.\mathbf{if}\ x \prec \alpha n\ \mathbf{then}\ \mathcal{R}^{\mathsf{o}}_{\prec} F(\bar{\alpha} n * x \mathbin{@} \beta)\big)$$

where $\prec$ is primitive recursive with $\mathsf{HA}^{\omega} \vdash \mathrm{TI}_{\prec}$.

Update Induction and Recursion. Before introducing update induction and recursion, we need to fix an encoding of "partial sequences". We do this by encoding partial objects of type ρ into objects of type $\mathbf{N} \times \rho$.[9] Whenever the first component is zero this indicates the value is "undefined" and if it is non-zero the value is the second component.[10] Let $\alpha\colon (\mathbf{N} \times \rho)^{\omega}$, then we call α a *partial sequence of type* ρ, and define the sequences $\mathsf{dom}(\alpha)\colon \mathbf{N}^{\omega}$ and $\mathsf{val}(\alpha)\colon \rho^{\omega}$ by

$$\mathsf{dom}(\alpha)n = \pi_0(\alpha n) \text{ and } \mathsf{val}(\alpha)n = \pi_1(\alpha n).$$

We also write $n \in \mathsf{dom}(\alpha)$ for $\mathsf{dom}(\alpha)n \neq 0$, $n \notin \mathsf{dom}(\alpha)$ for $\mathsf{dom}(\alpha)n = 0$, and $\alpha[n]$ for $\mathsf{val}(\alpha)n$.

For a partial sequence α of type ρ and $x\colon \rho$ we define the *update* α^x_n by

$$(\alpha^x_n)(m) := \begin{cases} \langle 1, x\rangle & \text{if } m = n, \\ \alpha(m) & \text{otherwise.} \end{cases}$$

Update induction is given by

$$\mathrm{UI}_U \qquad \forall\alpha\big(\forall n, x^{\rho}(n \notin \mathsf{dom}(\alpha) \to U(\alpha^x_n)) \to U(\alpha)\big) \to \forall\alpha U(\alpha),$$

where U ranges over open predicates of appropriate arity.

Update recursion is given by

$$\mathrm{UR}_{\rho}(\mathcal{R}^{\mathsf{u}}) \qquad \mathcal{R}^{\mathsf{u}} F\alpha =_{\mathbf{N}} F\alpha\big(\lambda n, x^{\rho}.\mathbf{if}\ n \notin \mathsf{dom}(\alpha)\ \mathbf{then}\ \mathcal{R}^{\mathsf{u}} F\alpha^x_n\big).$$

Extended Update Induction and Recursion. Whereas the update induction hypothesis allows only reference to one-point extensions, extended update induction allows reference to all (proper) extensions of a partial sequence.

For $x, y\colon \mathbf{N} \times \rho$ we define

$$x \ll y :\leftrightarrow \pi_0(x) = 0 \wedge \pi_0(y) \neq 0,$$

or in other words $x \ll y$ iff y is defined as a partial object and x is not.

Let $\alpha, \beta\colon (\mathbf{N} \times \rho)^{\omega}$ be partial sequences. Then we define $\alpha\{\beta\}\colon (\mathbf{N} \times \rho)^{\omega}$ by:

$$(\alpha\{\beta\})n := \begin{cases} \alpha n & \text{if } n \in \mathsf{dom}(\alpha), \\ \beta n & \text{otherwise.} \end{cases}$$

Extended update induction is given by

$$\mathrm{EUI}_U \qquad \forall\alpha\big(\forall n \forall\beta(\alpha n \ll \beta n \to U(\alpha\{\beta\})) \to U(\alpha)\big) \to \forall\alpha U(\alpha),$$

---

[9]Alternatively one could encode partial objects of type ρ into ρ itself, or use sum types instead.

[10]This notion of partiality should not be confused with the notion from Chapter 2. Here, by a case distinction on the first component, we can actually decide in the theory whether something is defined or not.

where U ranges over open predicates of appropriate arity. Note that $\alpha n \ll \beta n$ if and only if $n \in \mathsf{dom}(\beta)$ and $n \notin \mathsf{dom}(\alpha)$, i.e., n is a witness that $\alpha\{\beta\}$ is "more defined" than α.

Extended update recursion is given by

$$\mathrm{EUR}_\rho(\mathcal{R}^{\mathsf{e}}) \qquad \mathcal{R}^{\mathsf{e}} F \alpha^{(\mathbf{N}\times\rho)^\omega} =_{\mathbf{N}} F\alpha\,(\lambda n, \beta.\mathbf{if}\ \alpha n \ll \beta n\ \mathbf{then}\ \mathcal{R}^{\mathsf{e}} F(\alpha\{\beta\}))\,.$$

LEMMA 3.5. *In* $\mathsf{HA}^\omega[\mathcal{X}]$ *it is provable that* OI *implies* EUI*, and that* EUI *implies* UI.

PROOF. OI $\to$ EUI: Assume

$$\forall\alpha\big(\forall n \forall\beta(\alpha n \ll \beta n \to U(\alpha\{\beta\})) \to U(\alpha)\big). \tag{12}$$

We show: $\forall\alpha U(\alpha)$. Let $\prec$ be defined by $x \prec y$ iff $y \ll x$. Clearly $\mathrm{TI}_\prec$ is provable, so we can argue by OI. We have to show $U(\alpha)$ from the assumption

$$\forall\beta \prec_{\mathsf{lex}} \alpha\, U(\beta). \tag{13}$$

By (12) it suffices to show $U(\alpha\{\beta\})$ for all n and β with $\alpha n \ll \beta n$. Given n and β with $\alpha n \ll \beta n$, choose $m \le n$ minimal with $\alpha m \ll \beta m$. Then $\bar{\alpha} m = \overline{(\alpha\{\beta\})} m$ and $(\alpha\{\beta\}) m = \beta m \prec \alpha m$. Thus $\alpha\{\beta\} \prec_{\mathsf{lex}} \alpha$ and the claim follows from (13).

EUI $\to$ UI: Similar, using $\alpha_n^x(m) = (\alpha\{\alpha_n^x\})(m)$ for all m, and extensionality of open predicates. □

THEOREM 3.6. *The following principles are equivalent over* $\mathsf{PA}^\omega[\mathcal{X}]$*:*

(1) open induction OI,
(2) extended update induction EUI,
(3) update induction UI, *and*
(4) dependent choice DC.

PROOF. By the last lemma we only have to prove two directions.

UI $\to$ DC: Assume

$$\forall n \forall x^\rho \exists y^\rho A(n, x, y). \tag{14}$$

We have to show that there exists $f^{\mathbf{N}\to\rho}$ with $A(n, f(n), f(n+1))$ for all n. Call $\alpha^{(\mathbf{N}\times\rho)^\omega}$ a *partial choice function* if

$$0 \in \mathsf{dom}(\alpha) \land \forall n\,(n+1 \in \mathsf{dom}(\alpha) \to n \in \mathsf{dom}(\alpha) \land A(n, \alpha[n], \alpha[n+1]))\,.$$

Now define

$$U(\alpha) :\leftrightarrow \alpha \text{ is } \textit{not} \text{ a partial choice function.}$$

Then U is an open predicate. Clearly, any α with domain $\{0\}$ is a partial choice function, i.e., $\neg U(\alpha)$. By the contrapositive of UI, there exists a maximally defined partial choice function α, i.e.,

$$\neg U(\alpha) \land \forall n, x^\rho(n \notin \mathsf{dom}(\alpha) \to U(\alpha_n^x)).$$

It suffices to prove that α is total, i.e., $n \in \mathsf{dom}(\alpha)$ for all n, since then $f := \mathsf{val}(\alpha)$ is a choice function. Assume that α is not total; then there exists a least n with $n+1 \notin \mathsf{dom}(\alpha)$. By (14) there exists y such that $A(n, \alpha[n], y)$. Since n is minimal, we have $n \in \mathsf{dom}(\alpha)$ and therefore $\neg U(\alpha_{n+1}^y)$. But this contradicts the maximality of α.

DC $\to$ OI: We prove the contrapositive of OI, i.e.,

$$\exists\alpha\neg U(\alpha) \to \exists\alpha.\neg U(\alpha) \land \forall\beta \prec_{\mathsf{lex}} \alpha\, U(\beta).$$

Calling a sequence α *bad*, if $\neg U(\alpha)$ and *good* otherwise, this becomes the well-known minimal-bad-sequence argument of [**30**].

Define $B(s^{\rho^*}) := \exists\alpha \in s \neg U(\alpha)$ and assume that there is a bad sequence, i.e., $\exists\alpha\neg U(\alpha)$, i.e., $B(\langle\rangle)$. We have to show that there is (lexicographically) minimal bad sequence.

We now use the following variant of dependent choice which is derivable from DC_{ρ^*} (see the next lemma):

$$\forall s^{\rho^*}\exists x^\rho A(s,x) \to \exists\alpha^{\mathbf{N}\to\rho}\forall n A(\bar\alpha n, \alpha n). \tag{15}$$

We use (15) with

$$A(s,x) := \big(B(s) \to B(s*x) \wedge \forall y \prec x \neg B(s*y)\big).$$

We now prove $\forall s\exists x A(s,x)$. Using classical logic, it suffices to prove $B(s) \to \exists x.B(s*x) \wedge \forall y \prec x\neg B(s*y)$, for all s. So let s with $B(s)$ be given. We have to prove that there is a $\prec$-minimal x with $B(s*x)$. By the contraposition of $\mathrm{TI}_\prec$ (which is the minimum principle for $\prec$), it suffices to show $\exists x B(s*x)$. But, by the definition of $B(s)$, there is an $\alpha \in s$ with $\neg U(\alpha)$, and hence $B(s*\alpha|s|)$. This finishes the proof of $\forall s\exists x A(s,x)$.

So by (15), we obtain an α with

$$\forall n\big(B(\bar\alpha n) \to B(\bar\alpha(n+1)) \wedge \forall y \prec \alpha n \neg B(\bar\alpha n * y)\big). \tag{16}$$

Induction on n, and $B(\langle\rangle)$ yields $\forall n\, B(\bar\alpha n)$, i.e., $\forall n\exists\beta \in \bar\alpha n\, \neg U(\beta)$. By Remark 3.3 (2), we conclude $\neg U(\alpha)$, i.e., α is a bad sequence. It remains to prove that α is indeed a minimal bad sequence. Let $\beta \prec_{\mathsf{lex}} \alpha$; then there exists n with $\bar\alpha n = \bar\beta n$ and $\beta n \prec \alpha n$. Therefore (16) implies $\neg B(\bar\beta(n+1))$, that is $\forall\delta \in \bar\beta(n+1)\, U(\delta)$, in particular $U(\beta)$ what we had to prove. □

LEMMA 3.7. *The following is provable in* $\mathsf{HA}^\omega[\mathcal{X}] + \mathrm{DC}$*:*
(1) $\forall x^\rho\exists y^\rho B(x,y) \to \forall z^\rho\exists f^{\mathbf{N}\to\rho}\forall n.f0 = z \wedge B(f(n), f(n+1))$*, and*
(2) $\forall s^{\rho^*}\exists x^\rho A(s,x) \to \exists\alpha^{\mathbf{N}\to\rho}\forall n A(\bar\alpha n, \alpha n)$.

PROOF. For (1), see [**22**, p. 351 f.]. For (2), let $A(s^{\rho^*}, x^\rho)$ be given with $\forall s\exists x A(s,x)$. Define

$$B(s^{\rho^*}, t^{\rho^*}) :\leftrightarrow \big(t = s * \mathsf{last}(t) \wedge A(s, \mathsf{last}(t))\big),$$

where $\mathsf{last}(t) := t_{|t|-1}$ if $t \neq \langle\rangle$ and $\mathsf{last}(t) := 0^\rho$ otherwise. Using the assumption on A, one easily shows $\forall s\exists t B(s,t)$, and hence by (1) (for ρ^* instead of ρ and $z := \langle\rangle$), there exists an f with $f0 = \langle\rangle$ and for all n

$$f(n+1) = f(n) * \mathsf{last}(f(n+1)) \wedge A\big(f(n), \mathsf{last}(f(n+1))\big).$$

Putting $\alpha(n) = \mathsf{last}(f(n+1))$ one gets $\bar\alpha n = f(n)$ by induction on n. This yields $\forall n A(\bar\alpha n, \alpha n)$ as desired. □

THEOREM 3.8. *Both* $\mathbf{G}$ *and* $\mathbb{G}$ *satisfy* OR.

PROOF. We only have to prove $\mathbf{G} \models$ OR. Assume that $\prec$ is a primitive recursive binary relation with $\mathsf{HA}^\omega \vdash \mathrm{TI}_\prec$. Let D be a defined constant with the computation rule (additionally to the ones of Gödel's T)

$$DF\alpha \rhd F\alpha\big(\lambda n, x, \gamma.\textbf{if } x \prec \alpha n \textbf{ then } DF(\bar\alpha n * x \mathbin{@} \gamma)\big) : \mathbf{N},$$

where $DF\alpha\colon \mathbf{N}$. Define $\mathcal{R} := [\![D]\!] \in |\mathbf{C}|$. It is sufficient to prove that $\mathcal{R}$ is total. Let F be total. In order to prove that $\mathcal{R}F\alpha$ is total for all total α, we show that $\mathcal{R}F\alpha_\perp$ is total for all total α. This is sufficient, because if α is total, then $\alpha_\perp$ is total too and $\alpha_\perp \subseteq \alpha$. Hence $\mathcal{R}F\alpha_\perp \subseteq \mathcal{R}F\alpha$ and therefore $\mathcal{R}F\alpha$ is total whenever $\mathcal{R}F\alpha_\perp$ is total.

By Lemma 2.2, we have

$$\mathcal{R}F\alpha_\perp \text{ is total} \leftrightarrow \exists m.\mathcal{R}F(\bar{\alpha}m @ \perp) \text{ is total.} \tag{17}$$

Let X be a predicate variable of arity (ρ^*). We define an $\{X\}$-assignment Ξ by

$$\Xi(X) := \{s \in \mathbf{G}_{\rho^*} \mid \mathcal{R}F(s @ \perp) \text{ is total}\}.$$

For all total α we get by (17), $\mathbf{G}, \Xi \models \exists m\, X(\bar{\alpha}m)$ if and only if $\mathcal{R}F\alpha_\perp$ is total. Hence it is sufficient to prove $\mathbf{G}, \Xi \models \forall\alpha\exists m X(\bar{\alpha}m)$. We do this using open induction in $\mathbf{G}$ on the open predicate $U(\alpha) := X^\exists(\alpha) = \exists m\, X(\bar{\alpha}m)$. Note that open induction is valid in $\mathbf{G}$ by Theorem 3.6.

Let α be total and suppose the open induction hypothesis

$$\mathbf{G}, \Xi \models \forall n, x, \gamma\big(x \prec \alpha n \to X^\exists(\bar{\alpha}n * x @ \gamma)\big). \tag{18}$$

We have to prove $\mathbf{G}, \Xi \models X^\exists(\alpha)$, i.e., $\mathcal{R}F\alpha_\perp$ is total. By the definition of $\mathcal{R}$, we get

$$\mathcal{R}F\alpha_\perp = F\alpha_\perp([\![\lambda f, \beta\lambda n, x, \gamma.\mathbf{if}\ x \prec \beta(n)\ \mathbf{then}\ Df(\bar{\beta}n * x @ \gamma)]\!](F, \alpha_\perp)).$$

Because F and $\alpha_\perp$ are total it suffices to prove that $\mathcal{R}F(\overline{\alpha_\perp}n * x @ \gamma)$ is total for n, x, and γ total with $x \prec \alpha_\perp n$. Since n is total we conclude $\alpha(n) = \alpha_\perp(n)$ and therefore by (18), $\mathcal{R}F(\bar{\alpha}n * x @ \gamma)_\perp$ is total. Clearly, $(\bar{\alpha}n * x @ \gamma)_\perp \subseteq \overline{\alpha_\perp}n * x @ \gamma$, and hence monotonicity implies

$$\mathcal{R}F(\bar{\alpha}n * x @ \gamma)_\perp \subseteq \mathcal{R}F(\overline{\alpha_\perp}n * x @ \gamma).$$

Therefore, $\mathcal{R}F(\overline{\alpha_\perp}n * x @ \gamma)$ is total as well, which completes the proof. □

Analogously to the last proof, one can show that UR and EUR are valid in $\mathbf{G}$ and $\mathbb{G}$. This also follows from Theorem 4.3 below.

4. Definability

In this section we study the question whether one form of recursion scheme can be defined from the other. In order to do so, we first need to fix what we mean when we say that a functional defines another one.

For the rest of this chapter, we set $\mathcal{X} := \emptyset$.

Definition 4.1. Let $\mathcal{F}, \mathcal{G}$, and Δ be sets of closed HA^ω-formulas, such that all formulas in $\mathcal{F}$ and $\mathcal{G}$ are existential (i.e., of the form $\exists x A$ with A arbitrary). Then $\mathcal{F}$ *defines* $\mathcal{G}$ in Δ (or $\mathcal{G}$ is *definable* from $\mathcal{F}$ in Δ) if for each formula $\exists\Psi G(\Psi) \in \mathcal{G}$ there is a finite subset $\{\exists\Phi_1 F_1, \dots, \exists\Phi_n F_n\}$ of $\mathcal{F}$ and a closed term t such that

$$\mathsf{E}{-}\mathsf{HA}^\omega + \Delta + F_1 + \cdots + F_n \vdash G(t\Phi_1 \cdots \Phi_n).$$

If Δ is empty we just say that $\mathcal{F}$ defines $\mathcal{G}$. If $\mathcal{F}$ defines $\mathcal{G}$, and $\mathcal{G}$ defines $\mathcal{F}$, we say that they are *equivalent*.

Typically, $\mathcal{F}$ and $\mathcal{G}$ from the definition above are two of our schemes introduced in the last section and thus consist of formulas with a leading existential quantifier, followed by some universal quantifiers, and with an equation as kernel.

REMARK 4.2. If $\mathcal{F}$ defines $\mathcal{G}$ in Δ and $\mathbb{G} \models \Delta, \mathcal{F}$, then also $\mathbb{G} \models \mathcal{G}$.

THEOREM 4.3. OR *defines* EUR*, and* EUR *defines* UR*.*

PROOF. We first show that EUR is definable from OR. Define $\prec$ to be the inverse relation of $\ll$, i.e., $x \prec y$ iff $y \ll x$. Note that $\prec$ is well-founded. Let α, β be two partial sequences of type ρ and assume there exists n such that $\alpha n \ll \beta n$. So there exists a minimal $m \leq n$ such that $\alpha m \ll \beta m$, but then

$$\beta m \prec \alpha m \quad \text{and} \quad \bar{\alpha} m * \beta m \mathbin{@} (\alpha\{\beta\}) = \alpha\{\beta\},$$

so we have just showed how to define

$$\lambda n, \beta. \mathbf{if}\ \alpha n \ll \beta n\ \mathbf{then}\ G(\alpha\{\beta\})$$

in terms of

$$\lambda n, x, \gamma. \mathbf{if}\ x \prec \alpha n\ \mathbf{then}\ G(\bar{\alpha} n * x \mathbin{@} \gamma).$$

Thus it is clear that OR defines EUR.

It remains to prove that UR is definable from EUR. Let α be a partial sequence of type ρ and $n \notin \mathsf{dom}(\alpha)$. Then for any $x \colon \rho$,

$$\alpha n \ll \langle 1, x^\rho \rangle \quad \text{and} \quad \alpha^x_n = \alpha\{\bar{\alpha} n * \langle 1, x\rangle \mathbin{@} \alpha\}.$$

Now the claim follows as above. □

THEOREM 4.4. MBR *is definable by* UR*.*

PROOF. Given $H \colon \rho^* \to (\rho \to \mathbf{N}) \to \rho^\omega$ we define $\tilde{H}$ by

$$\tilde{H}(u^{(\mathbf{N}\times\rho)^*}, f) := H(\pi_1(u), f),$$

where $\pi_1(u)$ denotes componentwise right projection. Using UR we can define a term Φ satisfying

$$\Phi Y H \alpha^{(\mathbf{N}\times\rho)^\omega} =_{\mathbf{N}} Y(\lambda n. \beta(n)),$$

where $\beta(n) = \alpha[n]$ whenever $n \in \mathsf{dom}(\alpha)$, and otherwise

$$\beta(n) = \tilde{H}(\bar{\alpha} m, \lambda x^\rho. \Phi Y H \alpha^x_m, n),$$

with $m = \min_{k \leq n}(k \notin \mathsf{dom}(\alpha))$.[11]

Now we define Ψ by

$$\Psi Y H s := \Phi Y H (s' \mathbin{@} \bot),$$

where $s' := \langle\langle 1, s_0\rangle, \ldots, \langle 1, s_{|s|-1}\rangle\rangle \colon (\mathbf{N} \times \rho)^*$ and $\bot(n) := \langle 0, 0^\rho\rangle$. By extensionality we obtain

$$(s' \mathbin{@} \bot)^x_{|s|} = (s * x)' \mathbin{@} \bot.$$

[11] By extensionality, UR is equivalent to

$$\mathcal{R}^\circ F \alpha = F\alpha(\lambda n. \mathbf{if}\ n \notin \mathsf{dom}(\alpha)\ \mathbf{then}\ \lambda x. \mathcal{R}^\circ F \alpha^x_n).$$

Therefore, the reference to $\lambda x^\rho. \Phi Y H \alpha^x_m$ for fixed $m \notin \mathsf{dom}(\alpha)$ in the definition is unproblematic.

We now prove that Ψ satisfies the equation for modified bar recursion (where we omit the arguments Y and H):

$$\begin{aligned}\Psi s &= \Phi(s' \mathbin{@} \bot)\\ &\overset{(*)}{=} Y\big(s \mathbin{@} \tilde{H}(\overline{(s' \mathbin{@} \bot)}|s|, \lambda x^\rho.\Phi(s' \mathbin{@} \bot)^x_{|s|})\big)\\ &= Y\big(s \mathbin{@} \tilde{H}(s', \lambda x^\rho.\Phi((s * x)' \mathbin{@} \bot))\big)\\ &= Y\big(s \mathbin{@} H(s, \lambda x^\rho.\Psi(s * x))\big).\end{aligned}$$

At $(*)$ note that $n \in \mathsf{dom}(s' \mathbin{@} \bot)$ if and only if $n < |s|$. So if $n \notin \mathsf{dom}(s' \mathbin{@} \bot)$ then $n \geq |s|$ and hence $\min_{k \leq n}(k \notin \mathsf{dom}(s' \mathbin{@} \bot)) = |s|$. □

The next theorem is already implicitly indicated in [**4**, Section 6].

THEOREM 4.5. wBBC *is definable by* UR.

PROOF. With UR we can define a functional Φ such that

$$\Phi Y H \alpha =_{\mathbf{N}} Y\big(\lambda n.\mathbf{if}\ n \in \mathsf{dom}(\alpha)\ \mathbf{then}\ \alpha[n]\ \mathbf{else}\ H(n, \lambda x^\rho.\Phi Y H \alpha^x_n)\big),$$

where $\alpha\colon (\mathbf{N} \times \rho)^\omega$. For a list $s\colon (\mathbf{N} \times \rho)^*$ we define $s^\uparrow\colon (\mathbf{N} \times \rho)^\omega$ by

$$(\langle\rangle)^\uparrow(m) := \langle 0, 0^\rho\rangle, \text{ and } (s * \langle n, x\rangle)^\uparrow(m) := \begin{cases}\langle 1, x\rangle & \text{if } n = m,\\ s^\uparrow(m) & \text{otherwise.}\end{cases}$$

From extensionality we obtain $(s^\uparrow)^x_n = (s * \langle n, x\rangle)^\uparrow$. So, for $\Psi Y H s := \Phi Y H s^\uparrow$, we conclude

$$\begin{aligned}\Psi Y H s &= Y\big(\lambda n.\mathbf{if}\ n \in \mathsf{dom}(s^\uparrow)\ \mathbf{then}\ s^\uparrow[n]\ \mathbf{else}\ H(n, \lambda x^\rho.\Phi Y H (s^\uparrow)^x_n)\big)\\ &= Y\big(\lambda n.\mathbf{if}\ n \in \mathsf{dom}(s^\uparrow)\ \mathbf{then}\ s^\uparrow[n]\ \mathbf{else}\ H(n, \lambda x^\rho.\Phi Y H (s * \langle n, x\rangle)^\uparrow)\big)\\ &= Y\big(s\{\lambda n.H(n, \lambda x^\rho.\Psi Y H(s * \langle n, x\rangle))\}\big),\end{aligned}$$

where the last equality holds by extensionality. □

4.1. Equivalence of MBR **and** wMBR**.** Besides minor technical differences we follow [**7**] in this subsection.

LEMMA 4.6. MBR_ρ *is definable from* $\mathrm{wMBR}_{\rho^\omega}$*.*

PROOF. We first need some auxiliary functions. For $s = \langle s_0, \ldots, s_{n-1}\rangle$ of type ρ^* and $u = \langle u_0, \ldots, u_{m-1}\rangle$ of type $(\rho^\omega)^*$, we define

$$\begin{aligned}\mathsf{up}\, s &= \langle \lambda k s_0, \ldots, \lambda k s_{n-1}\rangle\colon (\rho^\omega)^*, \text{ and}\\ \mathsf{down}\, u &= \langle u_0(0), \ldots, u_{m-1}(0)\rangle\colon \rho^*.\end{aligned}$$

We claim:

(i) $\mathsf{down}(\mathsf{up}\, s) = s$,
(ii) $(\mathsf{up}\, s) * (\lambda k x) = \mathsf{up}(s * x)$,
(iii) $\lambda n.((\mathsf{up}\, s) \mathbin{@} \lambda k \alpha) n n = s \mathbin{@} \alpha$.

The first two statements are immediate; for (iii) let $n\colon \mathbf{N}$. If $n < |s| = |\mathsf{up}\, s|$, then

$$((\mathsf{up}\, s) \mathbin{@} \lambda k \alpha) n n = (\mathsf{up}\, s)_n n = (\lambda k s_n) n = s_n = (s \mathbin{@} \alpha) n.$$

Otherwise, if $n \geq |s|$, we get

$$((\mathsf{up}\, s) \mathbin{@} \lambda k \alpha) n n = (\lambda k \alpha) n n = \alpha n = (s \mathbin{@} \alpha) n.$$

So with extensionality the claim follows.

Now given functionals $Y\colon \rho^\omega \to \mathbf{N}$ and $H\colon \rho^* \to (\rho \to \mathbf{N}) \to \rho^\omega$ we define $\tilde{Y}\colon (\rho^\omega)^\omega \to \mathbf{N}$ and $\tilde{H}\colon (\rho^\omega)^* \to (\rho^\omega \to \mathbf{N}) \to \rho^\omega$ by

$$\tilde{Y}(\delta) = Y(\lambda n.\delta nn),$$
$$\tilde{H}(u, f) = H(\mathsf{down}\, u, \lambda x^\rho. f(\lambda k x)).$$

Now let Ψ with $\mathrm{wMBR}_{\rho^\omega}(\Psi)$ be given. We conclude:

$$\begin{aligned}
\Psi\tilde{Y}\tilde{H}(\mathsf{up}\, s) &= \tilde{Y}\big((\mathsf{up}\, s) @ \lambda k.\tilde{H}(\mathsf{up}\, s, \lambda\alpha^{\rho^\omega}.\Psi\tilde{Y}\tilde{H}((\mathsf{up}\, s) * \alpha))\big)\\
&= \tilde{Y}\big((\mathsf{up}\, s) @ \lambda k.H(\mathsf{down}(\mathsf{up}\, s), \lambda x^\rho.\Psi\tilde{Y}\tilde{H}((\mathsf{up}\, s) * (\lambda k x)))\big)\\
&= \tilde{Y}\big((\mathsf{up}\, s) @ \lambda k.H(s, \lambda x^\rho.\Psi\tilde{Y}\tilde{H}(\mathsf{up}(s * x)))\big)\\
&= Y\big(\lambda n.((\mathsf{up}\, s) @ \lambda k.H(s, \lambda x^\rho.\Psi\tilde{Y}\tilde{H}(\mathsf{up}(s * x))))nn\big)\\
&\overset{iii}{=} Y\big(s @ H(s, \lambda x^\rho.\Psi\tilde{Y}\tilde{H}(\mathsf{up}(s * x)))\big).
\end{aligned}$$

Hence $\mathrm{MBR}_\rho(\lambda YHs.\Psi\tilde{Y}\tilde{H}(\mathsf{up}\, s))$. □

COROLLARY 4.7. *Modified bar recursion and weak modified bar recursion are equivalent.*

4.2. Definability of SBR **by** MBR**.** In this subsection we show how Spector's bar recursion can be defined by a nested use of modified bar recursion. First we need the following minimization functional:

DEFINITION 4.8. $\tilde{\mu}(Y, \alpha) = \min_{n \geq 0}(Y(\overline{\alpha, n}) < n)$.

It was observed by Howard in [**21**, Lemma 3C, p. 113] that $\tilde{\mu}$ is definable by SBR, more precisely by $\mathrm{SBR}_{\rho,\mathbf{N}}$ if ρ^ω is the type of α in the definition above. We will now see that $\tilde{\mu}$ is also definable by modified bar recursion. For this, we use the same idea as in [**7**] but our implementation and verification proof is different. Lemma 4.13 and 4.14 are from loc. cit. but we have filled some gaps which needed bar induction for verification.

First, we need the following auxiliary functional:

LEMMA 4.9. *The following recursion scheme is definable from* MBR*:*

$$\Phi YH\alpha n =_{\mathbf{N}} Y\big(\bar{\alpha}n @ H(\alpha, n, \Phi YH\alpha(n+1))\big).$$

PROOF. For $H\colon \rho^\omega \to \mathbf{N} \to \mathbf{N} \to \rho^\omega$ and $\alpha\colon \rho^\omega$ define

$$H_\alpha(s, f^{\rho\to\mathbf{N}}) := H(\alpha, |s|, f(\alpha|s|)).$$

Suppose $\mathrm{MBR}_\rho(\Psi)$ and define $\Phi YH\alpha n := \Psi YH_\alpha(\bar{\alpha}n)$. By the defining equations for Ψ and H_α, we conclude:

$$\begin{aligned}
\Phi YH\alpha n &= \Psi YH_\alpha(\bar{\alpha}n)\\
&= Y\big(\bar{\alpha}n @ H_\alpha(\bar{\alpha}n, \lambda x^\rho.\Psi YH_\alpha(\bar{\alpha}n * x))\big)\\
&= Y\big(\bar{\alpha}n @ H(\alpha, n, \Psi YH_\alpha(\bar{\alpha}n * \alpha n))\big)\\
&= Y\big(\bar{\alpha}n @ H(\alpha, n, \Phi YH\alpha(n+1))\big).
\end{aligned}$$
□

LEMMA 4.10. *The minimization functional $\tilde{\mu}$ is definable from* MBR*.*

Proof. The proof is based on the following (informal) observation. Let $n := \tilde{\mu}(Y, \alpha)$ then $n > 0$. By the minimality of n, we know that $Y(\overline{\alpha, n-1}) \geq n - 1$ and hence

$$Y(\overline{\alpha, n-1}) + 1 \geq n.$$

Therefore we can use $Y(\overline{\alpha, n-1}) + 1$ as an upper bound for a bounded search for n. So our main task is to define $Y(\overline{\alpha, n-1})$ using MBR.

Let $\tilde{\mu}^b$ be the bounded variant of $\tilde{\mu}$, i.e., $\tilde{\mu}^b(Y, \alpha, m)$ is the least n with $n \leq m$ such that $Y(\overline{\alpha, n}) < n$ if such an n exists, and $m + 1$ otherwise. Clearly, $\tilde{\mu}^b$ is definable by primitive recursion and always positive.

For readability we write $(\overline{\overline{\alpha}}, n)$ for $(\overline{\alpha, n})$. We define the functional Φ using the scheme from Lemma 4.9 (and hence MBR) by

$$\Phi(Y, \alpha, n) = Y\big(\bar{\alpha}n \mathbin{@} (\overline{\overline{\alpha}}, \tilde{\mu}^b\,(Y, \alpha, \Phi(Y, \alpha, n+1) + 1) - 1)\big).$$

Now we set

$$\tilde{\mu}(Y, \alpha) := \tilde{\mu}^b(Y, \alpha, \Phi(Y, \alpha, 0) + 1).$$

Fix Y and α. For readability we omit the arguments Y and α of $\tilde{\mu}$, $\tilde{\mu}^b$, and Φ. By the definition of $\tilde{\mu}$ and $\tilde{\mu}^b$ we obtain

$$\tilde{\mu} > \tilde{\mu}^b(n) \to \tilde{\mu}^b(n) = n + 1. \tag{19}$$

In order to prove that our definition of $\tilde{\mu}$ is correct, it suffices to show that the search is successful, i.e., $\tilde{\mu} \leq \Phi(0) + 1$. Assume the contrary, i.e.,

$$\tilde{\mu} > \Phi(0) + 1. \tag{20}$$

We now prove $\tilde{\mu} > \Phi(n) + n + 1$ by induction on n. Of course, this gives the desired contradiction for $n = \tilde{\mu}$. In the case $n = 0$ this follows from (20). In the induction step $n \to n + 1$ the IH is $\tilde{\mu} > \Phi(n) + n + 1$. We now claim

$$\Phi(n) + 1 \geq \tilde{\mu}^b(\Phi(n+1) + 1). \tag{21}$$

Case $\tilde{\mu}^b(\Phi(n{+}1){+}1){-}1 < n$. By IH we have $n < \tilde{\mu}$ and thus $Y(\overline{\alpha, n}) \geq n$. We conclude:

$$\Phi(n) + 1 = Y(\overline{\alpha, n}) + 1 \geq n + 1 > \tilde{\mu}^b(\Phi(n+1) + 1).$$

Case $\tilde{\mu}^b(\Phi(n+1) + 1) - 1 \geq n$. Then by the definition of Φ:

$$\Phi(n) + 1 = Y\big(\overline{\overline{\alpha}}, \underbrace{\tilde{\mu}^b(\Phi(n+1) + 1)}_{=:S} - 1\big) + 1 =: R.$$

If the search S is successful, we obtain

$$R \geq \tilde{\mu}^b(\Phi(n+1) + 1) - 1 + 1 = \tilde{\mu}^b(\Phi(n+1) + 1).$$

Otherwise, if S is not successful, we have

$$\begin{aligned} R &= Y\big(\overline{\overline{\alpha}}, \Phi(n+1) + 2 - 1\big) + 1 = Y\big(\overline{\overline{\alpha}}, \Phi(n+1) + 1\big) + 1 \\ &\geq \Phi(n+1) + 1 + 1 = \tilde{\mu}^b(\Phi(n+1) + 1). \end{aligned}$$

Thus we have proved $\Phi(n) + 1 \geq \tilde{\mu}^b(\Phi(n+1) + 1)$.

We conclude:

$$\begin{aligned}\tilde{\mu} &> \Phi(n) + n + 1 && \text{by IH}\\ &\geq \tilde{\mu}^b(\Phi(n+1)+1) + n && \text{see above}\\ &= \Phi(n+1) + (n+1) + 1 && \text{by (19)}\end{aligned}$$

This completes the proof. $\square$

REMARK 4.11. Assuming extensionality and the following weak continuity property

$$\forall Y^{\rho^\omega \to \mathbf{N}} \forall \alpha \exists n \, (Y(\overline{\alpha, n}) < n)\,, \tag{22}$$

then $\tilde{\mu}$ is primitive recursive.

PROOF. Given Y and α, define

$$\beta_{Y,\alpha}(n) := \beta(n) := \begin{cases} 0 & \text{if } \exists m \leq n+1. Y(\overline{\alpha, m}) < m, \\ \alpha(n) & \text{otherwise.} \end{cases}$$

By (22) there exists n such that $Y(\overline{\alpha, n}) < n$. Since the formula $Y(\overline{\alpha, n}) < n$ is quantifier-free we may assume that n is minimal. By construction and extensionality we know that $\beta = (\overline{\alpha, n-1})$. The minimality of n yields $Y(\beta) = Y(\overline{\alpha, n-1}) \geq n-1$ and therefore $\lambda Y, \alpha. \tilde{\mu}^b(Y, \alpha, Y(\beta_{Y,\alpha})+1)$ satisfies the defining equation for $\tilde{\mu}$. $\square$

REMARK 4.12. $\tilde{\mu}$ can be generalized to $\tilde{\mu}'$ with

$$\tilde{\mu}'(Y, \alpha, k) = \min_{n \geq k} (Y(\overline{\alpha, n}) < n).$$

An easy calculation shows that we can put

$$\tilde{\mu}'(Y, \alpha, k) := \tilde{\mu}(\lambda \beta. \max\{Y(\bar{\alpha}k \mathbin{@} \beta), k \dot{-} 1\}, \alpha).$$

LEMMA 4.13. $\mathrm{SBR}_{\rho,\mathbf{N}}$ *is definable by* MBR *and* $\tilde{\mu}$ *using* BI'_D.

PROOF. Before we begin with the proof, let us fix the following notation. For $n\colon \mathbf{N}$ let $n^\uparrow := \mathbf{emb}_\rho(n)$ denote the canonical embedding of $\mathbf{N}$ into ρ, and for $x\colon \rho$ let $x^\downarrow := \psi_\rho(x)\colon \mathbf{N}$ be the inverse operation. For a list $s = \langle s_0, \ldots, s_{n-1} \rangle\colon \rho^*$ we write $\langle x^\sigma, s \rangle$ for the list $\langle \langle x, s_0 \rangle, \ldots, \langle x, s_{n-1}, \rangle \rangle\colon (\sigma \times \rho)^*$. Projections are extended to lists and sequences componentwise. We use α and β as variables of type $(\mathbf{N} \times \rho)^\omega$.

We want to define the functional Φ satisfying the equation

$$\Phi Y G H s =_{\mathbf{N}} \begin{cases} G(s) & \text{if } Y(\hat{s}) < |s|, \\ H(s, \lambda x^\rho. \Phi Y G H(s * x)) & \text{otherwise.} \end{cases} \tag{23}$$

Given Y, G, and H (of appropriate type) we define $\tilde{H}$ and $\tilde{Y}_{G,k}$ by

$$\tilde{H}(u^{(\mathbf{N}\times\rho)^*}, f^{(\mathbf{N}\times\rho)\to\rho}, m) =_{\mathbf{N}\times\rho} \langle 1, [H(\pi_1(u), \lambda x^\rho. f\langle 0, x\rangle)]^\uparrow \rangle$$

and

$$\tilde{Y}_{G,k}(\alpha) =_{\mathbf{N}} \begin{cases} G(\pi_1(\bar{\alpha}n)) & \text{if } \forall i < n \, \pi_0(\alpha i) = 0, \\ (\pi_1(\alpha n))^\downarrow & \text{otherwise.} \end{cases} \tag{24}$$

where $n = \tilde{\mu}'(Y, \pi_1(\alpha), k)$. First note that:

$$Y(\hat{s}) \geq |s| \to \forall\beta.\tilde{Y}_{G,|s|}(\langle 0, s\rangle * \beta) = \tilde{Y}_{G,|s|+1}(\langle 0, s\rangle * \beta). \tag{25}$$

To see this assume $Y(\hat{s}) \geq |s|$, then for $\alpha := \langle 0, s\rangle * \beta$ we get $\tilde{\mu}'(Y, \pi_1(\alpha), |s|) > |s|$, and therefore $\tilde{\mu}'(Y, \pi_1(\alpha), |s|) = \tilde{\mu}'(Y, \pi_1(\alpha), |s| + 1)$. Hence both sides in the conclusion of (25) are equal.

Now assume $\mathrm{MBR}_{\mathbf{N}\times\rho}(\Psi)$ and define

$$\Phi Y H s =_{\mathbf{N}} \Psi(\tilde{Y}_{G,|s|}, \tilde{H}, \langle 0, s\rangle).$$

We now have to verify (23). By unfolding the definitions, we get:

$$\begin{aligned}\Phi Y H s &= \Psi(\tilde{Y}_{G,|s|}, \tilde{H}, \langle 0, s\rangle)\\ &= \tilde{Y}_{G,|s|}\big(\langle 0, s\rangle \mathbin{@} \tilde{H}(\langle 0, s\rangle, \lambda p^{\mathbf{N}\times\rho}.\Psi(\tilde{Y}_{G,|s|}, \tilde{H}, \langle 0, s\rangle * p))\big)\\ &= \tilde{Y}_{G,|s|}\big(\langle 0, s\rangle \mathbin{@} \lambda m\langle 1, [H(s, \lambda x^{\rho}.\Psi(\tilde{Y}_{G,|s|}, \tilde{H}, \langle 0, s * x\rangle))]^{\uparrow}\rangle\big).\end{aligned}$$

In the case $Y(\hat{s}) < |s|$, we have

$$\begin{aligned}\Phi Y H s &= \tilde{Y}_{G,|s|}(\langle 0, s\rangle \mathbin{@} \dots)\\ &= G(\pi_1(\langle 0, s\rangle)) && \text{by (24) since } n = |s|\\ &= G(s).\end{aligned}$$

Now assume $Y(\hat{s}) \geq s$. Define $Q(t)$ for $t\colon \rho^*$ to be

$$\Psi(\tilde{Y}_{G,|s|}, \tilde{H}, \langle 0, s * t\rangle) = \Psi(\tilde{Y}_{G,|s|+1}, \tilde{H}, \langle 0, s * t\rangle). \tag{26}$$

First, we prove $Q(t)$ for all t using BI'_D with $P(t)$ defined as $Y(\widehat{s * t}) < |s*t|$. Let us verify the premises of BI'_D: That P is an infinite bar follows because of the presence of $\tilde{\mu}'$. Now assume $P(t)$; we show $Q(t)$. We get

$$\Psi(\tilde{Y}_{G,|s|}, \tilde{H}, \langle 0, s * t\rangle) = Y_{G,|s|}(\langle 0, s * t\rangle \mathbin{@} \dots) = G(s * t'),$$

where t' is the shortest initial segment of t with $Y(\widehat{s * t'}) < |s * t'|$. By our assumption on s we know $t' \neq \langle\rangle$. Accordingly, a similar calculation shows

$$\Psi(\tilde{Y}_{G,|s|+1}, \tilde{H}, \langle 0, s * t\rangle) = G(s * t').$$

This completes the proof of $P(t) \to Q(t)$. Now assume $Q(t * x)$ for all x; we must show $Q(t)$. By assumption

$$\lambda x^{\rho}.\Psi(\tilde{Y}_{G,|s|}, \tilde{H}, \langle 0, s * (t * x)\rangle) = \lambda x^{\rho}.\Psi(\tilde{Y}_{G,|s|+1}, \tilde{H}, \langle 0, s * (t * x)\rangle).$$

This gives

$$\begin{aligned}&\Psi(\tilde{Y}_{G,|s|}, \tilde{H}, \langle 0, s * t\rangle)\\ &= \tilde{Y}_{G,|s|}\big(\langle 0, s * t\rangle \mathbin{@} \lambda m\langle 1, [H(s * t, \lambda x^{\rho}.\Psi(\tilde{Y}_{G,|s|}, \tilde{H}, \langle 0, s * (t * x)\rangle))]^{\uparrow}\rangle\big)\\ &= \tilde{Y}_{G,|s|}\big(\langle 0, s * t\rangle \mathbin{@} \lambda m\langle 1, [H(s * t, \lambda x^{\rho}.\Psi(\tilde{Y}_{G,|s|+1}, \tilde{H}, \langle 0, s * (t * x)\rangle))]^{\uparrow}\rangle\big)\\ &\overset{(25)}{=} \tilde{Y}_{G,|s|+1}\big(\langle 0, s * t\rangle \mathbin{@} \lambda m\langle 1, [H(s * t, \lambda x^{\rho}.\Psi(\tilde{Y}_{G,|s|+1}, \tilde{H}, \langle 0, s * t * x\rangle))]^{\uparrow}\rangle\big)\\ &= \Psi(\tilde{Y}_{G,|s|+1}, \tilde{H}, \langle 0, s * t\rangle).\end{aligned}$$

This finishes the proof of $\forall t\, Q(t)$.

Finally, let us calculate Φ:

$$\begin{aligned}\Phi Y H s &= \tilde{Y}_{G,|s|}\big(\langle 0,s\rangle @ \lambda m\langle 1,[H(s,\lambda x^\rho.\Psi(\tilde{Y}_{G,|s|},\tilde{H},\langle 0,s*x\rangle))]^{\uparrow}\rangle\big)\\ &\overset{(24)}{=} \big[H(s,\lambda x^\rho.\Psi(\tilde{Y}_{G,|s|},\tilde{H},\langle 0,s*x\rangle))\big]^{\uparrow\downarrow}\\ &\overset{(26)}{=} H(s,\lambda x^\rho.\Psi(\tilde{Y}_{G,|s|+1},\tilde{H},\langle 0,s*x\rangle))\\ &= H(s,\lambda x^\rho.\Phi YGH(s*x)).\end{aligned}$$

This is what we had to show. □

LEMMA 4.14. SBR *is definable from* $\{\mathrm{SBR}_{\rho,\mathbf{N}} \mid \rho \in \mathsf{Ty}\}$ *using* BI'_D.

PROOF SKETCH. We only show that $\mathrm{SBR}_{\rho\times\tau,\mathbf{N}}$ defines $\mathrm{SBR}_{\rho,\tau\to\mathbf{N}}$. The rest follows using the usual primitive recursive coding machinery.

We want to define a functional Φ satisfying

$$\Phi YGHs =_{\tau\to\mathbf{N}} \begin{cases} G(s) & \text{if } Y(\hat{s}) < |s|,\\ H(s,\lambda x^\rho.\Phi YGH(s*x)) & \text{otherwise.}\end{cases} \tag{27}$$

Given Y, G, and H we define (with $\alpha\colon (\rho\times\tau)^\omega$, $u\colon (\rho\times\tau)^*$, and $f\colon (\rho\times\tau)\to\mathbf{N}$):

$$\begin{aligned} Y'(\alpha) &:= Y\big(\lambda n.\pi_0(\alpha(n+1))\big)+1,\\ G'(u) &:= G\big(\underline{\pi}_0(u),\pi_1(u_{|u|-1})\big),\\ H'(u,f) &:= H\big(\underline{\pi}_0(u),\lambda x^\rho\lambda z^\tau.f\langle x,z\rangle,\pi_1(u_{|u|-1})\big),\end{aligned}$$

where $\underline{\pi}_0\langle u_0,\dots,u_{|u|-1}\rangle := \langle \pi_0(u_1),\dots,\pi_0(u_{|u|-1})\rangle$ and we stipulate $s_x^{\rho^*} := 0^\rho$ for $x<0$.

Moreover, for $s\colon \rho^*$, $t\colon \tau^*$, and $z\colon \tau$ we define

$$\begin{aligned} \langle\!\langle s,z\rangle\!\rangle &:= \langle\langle 0^\rho,z\rangle,\langle s_0,z\rangle,\dots,\langle s_{|s|-1},z\rangle\rangle,\\ \langle s,t\rangle &:= \langle\langle s_0,t_0\rangle,\dots,\langle s_{|s|-1},t_{|s|-1}\rangle\rangle.\end{aligned}$$

Then we have $\underline{\pi}_0\langle\!\langle s,z\rangle\!\rangle = s$ and $\pi_1(\langle\!\langle s,z\rangle\!\rangle_{|\langle\!\langle s,z\rangle\!\rangle|-1}) = z$. Also note that:

$$Y(\hat{s}) < |s| \leftrightarrow Y'(\langle\!\langle s,z\rangle\!\rangle @ 0) < |\langle\!\langle s,z\rangle\!\rangle|.$$

Now let Ψ with $\mathrm{SBR}_{\rho\times\tau,\mathbf{N}}(\Psi)$ be given. We now verify that

$$\lambda YGHsz.\Psi Y'G'H'\langle\!\langle s,z\rangle\!\rangle$$

satisfies Equation (27). In the case $Y(\hat{s}) < |s|$ we get

$$\Psi'\langle\!\langle s,z\rangle\!\rangle := \Psi Y'G'H'\langle\!\langle s,z\rangle\!\rangle = G'\langle\!\langle s,z\rangle\!\rangle = G(s,z).$$

And in case $Y(\hat{s}) \geq |s|$ we have

$$\begin{aligned}\Psi'\langle\!\langle s,z\rangle\!\rangle &= H'\left(\langle\!\langle s,z\rangle\!\rangle,\lambda y^{\rho\times\tau}.\Psi'(\langle\!\langle s,z\rangle\!\rangle * y)\right)\\ &= H\left(s,\lambda x^\rho\lambda z_0^\tau.\Psi'(\langle\!\langle s,z\rangle\!\rangle * \langle x,z_0\rangle),z\right).\end{aligned}$$

So it is enough to show $\Psi'\langle\!\langle s,z\rangle\!\rangle * \langle x,z_0\rangle = \Psi'\langle\!\langle s*x,z_0\rangle\!\rangle$. This follows from

$$Q(s) := \forall t,t'^{\tau^*}\left(s\neq\langle\rangle \wedge t_{|s|-1} = t'_{|s|-1} \to \Psi'\langle s,t\rangle = \Psi'\langle s,t'\rangle\right)$$

for all s. We prove this by BI'_D with the decidable and infinite bar $P(s) := Y(\hat{s}) < |s|$. That P is an infinite bar follows from the fact that $\mathrm{SBR}_{\rho,\mathbf{N}}$

defines $\tilde{\mu}$ (cf. the beginning of this subsection) and thus also $\tilde{\mu}'$. Proving that $P(s) \to Q(s)$ and $\forall x Q(s * x) \to Q(s)$ for all s is easy. □

COROLLARY 4.15. SBR *is definable from* MBR *using* BI'_D.

PROOF. Combine the last three lemmas. □

4.3. Auxiliary Definitions. For the next section we need a slight generalization of modified bar recursion.

LEMMA 4.16. MBR *is equivalent to each of the following schemes:*
(1) $\Psi Y H s =_{\mathbf{N}} Y\big(s * H(s, \lambda x^\rho.\Psi Y H(s * x))\big)$,
(2) $\Psi Y H s =_{\rho^\omega} s \mathbin{@} H\big(s, \lambda x^\rho.Y(\Psi Y H(s * x))\big)$.

PROOF. Easy. □

LEMMA 4.17. *Over* $\mathrm{BI}'_D + \mathrm{Cont}$, MBR *defines the following scheme:*

$$\Psi Y H s^{\rho^*} =_{\mathbf{N}} Y\big(s * H(s, \lambda x^\rho \lambda t^{\rho^*}.\Psi Y H(s * t * x))\big). \tag{28}$$

Moreover, (28) *is equivalent to each of the following schemes:*

$$\Psi Y H s^{\rho^*} =_{\mathbf{N}} Y\big(s \mathbin{@} H(s, \lambda x^\rho \lambda t^{\rho^*}.\Psi Y H(s * t * x))\big), \tag{29}$$

$$\Psi Y H s^{\rho^*} =_{\rho^\omega} s \mathbin{@} H(s, \lambda x^\rho \lambda t^{\rho^*}.Y(\Psi Y H(s * t * x)))). \tag{30}$$

PROOF. The equivalences of (28) with (29) and with (30) are easy to see. We only define (28) using MBR_{ρ^*} based on the argument sketched in [**6**, Lemma 2]. For this, we first need some auxiliary functionals for lists of lists. We use the following type conventions: $\delta\colon (\rho^*)^\omega$, $u, v, w\colon \rho^{**}$, $s, t\colon \rho^*$, and $\alpha\colon \rho^\omega$. Now let us define functionals $\mathsf{melt}\colon \rho^{**} \to \rho^*$, $\mathsf{freeze}\colon \rho^* \to \rho^{**}$ and $\mathsf{flat}\colon (\rho^*)^\omega \to \rho^\omega$ by

$$\begin{aligned} \mathsf{melt}\langle u_0, \dots, u_{n-1}\rangle &:= u_0 * \dots * u_{n-1},\\ \mathsf{freeze}\langle s_0, \dots, s_{n-1}\rangle &:= \langle\langle s_0\rangle, \dots, \langle s_{n-1}\rangle\rangle,\\ \mathsf{flat}\,\delta n &:= \big(\mathsf{melt}(\bar{\delta}(n+1))\big)_n. \end{aligned}$$

We also extend freeze to sequences α by $\mathsf{freeze}\,\alpha n := \langle \alpha n\rangle$. Moreover, define

$$\langle\rangle \notin u :\leftrightarrow \forall n < |u|\, u_n \neq \langle\rangle, \text{ and } \langle\rangle \notin \delta :\leftrightarrow \forall n\, \delta n \neq \langle\rangle.$$

Clearly, $\mathsf{melt}(\mathsf{freeze}\, s) = s$. We now claim the following:
(i) $s \neq \langle\rangle \wedge \langle\rangle \notin \delta \to \mathsf{flat}(\langle s\rangle * \delta) = s * \mathsf{flat}(\delta)$,
(ii) $\langle\rangle \notin u \wedge \langle\rangle \notin \delta \to \mathsf{flat}(u * \delta) = (\mathsf{melt}\, u) * \mathsf{flat}(\delta)$.

(ii) follows easily by induction on u using (i). It remains to prove (i). Let $s = \langle s_0, \dots, s_n\rangle$ (i.e., $|s| = n+1$), $\langle\rangle \notin \delta$, and $k\colon \mathbf{N}$. We show: $\mathsf{flat}(\langle s\rangle * \delta)k = (s * \mathsf{flat}(\delta))k$.

Case $k \leq n$. Then:

$$\mathsf{flat}(\langle s\rangle * \delta)k = \big(\mathsf{melt}(\overline{(\langle s\rangle * \delta)}(k+1))\big)_k = \big(\langle s_o, \dots, s_n, \dots\rangle\big)_k = s_k.$$

And $s_k = (s * \mathsf{flat}(\delta))k$ (for $k = 0$ this holds since $s \neq \langle\rangle$).

Case $k > n$. Then:

$$\begin{aligned} \mathsf{flat}(\langle s\rangle * \delta)(k) &= \big(\mathsf{melt}(\overline{(\langle s\rangle * \delta)}(k+1))\big)_k\\ &= (s * \delta 0 * \ldots * \delta(k-1))_k\\ &= (\delta 0 * \ldots * \delta(k-1))_{k-|s|}, \end{aligned}$$

and

$$\begin{aligned}(s * \mathsf{flat}(\delta))k &= \mathsf{flat}\,\delta(k - |s|)\\ &= \big(\mathsf{melt}(\bar{\delta}(k-n))\big)_{k-|s|}\\ &= (\delta 0 * \ldots * \delta(k-|s|))_{k-|s|}\,.\end{aligned}$$

So the claim follows from $\langle\rangle \notin \delta$. This completes the proof of (i).

Note that we have $\langle\rangle \notin \mathsf{freeze}\,s$, $\langle\rangle \notin \mathsf{freeze}\,\alpha$, and $\mathsf{flat}(\mathsf{freeze}\,\alpha) = \alpha$.

Now given $Y^{\rho^\omega \to \mathbf{N}}$ and $H^{\rho^* \to (\rho \to \rho^* \to \mathbf{N}) \to \rho^\omega}$ define

$$\begin{aligned}H'(u, F^{\rho^* \to \mathbf{N}}) &:=_{(\rho^*)^\omega} \mathsf{freeze}(H(\mathsf{melt}\,u, \lambda x^\rho \lambda t^{\rho^*}.F(t * x))),\\ Y'(\delta) &:=_{\mathbf{N}} Y(\mathsf{flat}\,\delta).\end{aligned}$$

Let Φ' satisfy Equation (1) from Lemma 4.16 at type ρ^*. Define $\Phi Y H u := \Phi' Y' H' u$. For readability we omit the arguments Y and H from Φ. By (ii), we conclude for $\langle\rangle \notin u$:

$$\begin{aligned}\Phi(u) &= Y'\big(u * H'(u, \lambda t^{\rho^*}.\Phi(u * t))\big)\\ &= Y\Big(\mathsf{flat}\big[u * \mathsf{freeze}\big(H\big(\mathsf{melt}\,u, \lambda x^\rho \lambda t^{\rho^*}.\Phi(u * (t * x))\big)\big)\big]\Big)\\ &\overset{ii}{=} Y\Big(\mathsf{melt}(u) * \mathsf{flat}\big(\mathsf{freeze}\big(H\big(\mathsf{melt}\,u, \lambda x^\rho \lambda t^{\rho^*}.\Phi(u * (t * x))\big)\big)\big)\Big)\\ &= Y\big(\mathsf{melt}(u) * H\big(\mathsf{melt}\,u, \lambda x^\rho \lambda t^{\rho^*}.\Phi(u * (t * x))\big)\big).\end{aligned} \tag{31}$$

For u and $i < |u|$ one easily defines $u_{(i)}$ and $u^{(i)}$ such that $|u_{(i)}| = i$ and $u = u_{(i)} * u^{(i)}$. Let

$$Q(u) :\leftrightarrow \Big(\langle\rangle \notin u \to \forall i < |u|.\Phi(u) = \Phi\big((\mathsf{freeze}(\mathsf{melt}\,u_{(i)})) * u^{(i)}\big)\Big).$$

We now prove $\forall u\, Q(u)$ using BI'_D (with $P := Q$) and Cont. Clearly, $Q(u)$ is decidable for all u. We now verify the remaining premises of BI'_D.

(1) $\forall \delta \forall k \exists n \geq k\, Q(\bar{\delta} n)$. Let δ and k be given and let n be a point of continuity of Y at $\mathsf{flat}\,\delta$. W.l.o.g. we can assume $n \geq k$. Suppose $\langle\rangle \notin \bar{\delta} n$ and $i < n$. We have to show

$$\Phi(\bar{\delta} n) = \Phi\big((\mathsf{freeze}(\mathsf{melt}\,\bar{\delta} i)) * \langle \delta(i), \ldots, \delta(n-1)\rangle\big).$$

Observe that

$$\mathsf{melt}\big((\mathsf{freeze}(\mathsf{melt}\,\bar{\delta} i)) * \langle \delta(i), \ldots, \delta(n-1)\rangle\big) = \mathsf{melt}\,\bar{\delta} n.$$

Using this and (31), we calculate:

$$\begin{aligned}&\Phi\big((\mathsf{freeze}(\mathsf{melt}\,\bar{\delta} i)) * \langle \delta(i), \ldots, \delta(n-1)\rangle\big)\\ &= Y\big((\mathsf{melt}\,\bar{\delta} n) * \ldots\big)\\ &= Y\big((\mathsf{melt}\,\bar{\delta} n) * H(\mathsf{melt}\,\bar{\delta} n, \lambda x^\rho \lambda t.\Phi(\bar{\delta} n * (t * x)))\big)\\ &= \Phi(\bar{\delta} n),\end{aligned}$$

where in the second equality, we have used that n is a point of continuity of Y at $\mathsf{flat}\,\delta$, and $(\mathsf{melt}\,\bar{\delta} n) * \alpha \in (\overline{\mathsf{flat}\,\delta}) n$ for any α since $\langle\rangle \notin \bar{\delta} n$.

(2) Let u and i be given with $\forall t\, Q(u * t)$, $\langle\rangle \notin u$, and $i < |u|$. Define $v := u_{(i)}$ and $w := u^{(i)}$. We show: $\Phi(u) = \Phi\big((\mathsf{freeze}(\mathsf{melt}\, v)) * w\big)$. We have $\mathsf{melt}(u) = \mathsf{melt}\big((\mathsf{freeze}(\mathsf{melt}\, v)) * w\big)$ and thus by (31):

$$\begin{aligned}
&\Phi\big((\mathsf{freeze}(\mathsf{melt}\, v)) * w\big) \\
&= Y\Big(\mathsf{melt}(u) * H\big(\,\mathsf{melt}\, u, \lambda x^\rho \lambda t.\Phi\big(((\mathsf{freeze}(\mathsf{melt}\, v)) * w) * (t * x)\big)\big)\Big) \\
&= Y\big(\mathsf{melt}(u) * H(\mathsf{melt}\, u, \lambda x^\rho \lambda t.\Phi(u * (t * x)))\big) \\
&= \Phi(u),
\end{aligned}$$

where in the second step, we have used the assumption $Q(u*(t*x))$. This finishes the proof of $\forall u\, Q(u)$. In particular, $\Phi(u) = \Phi(\mathsf{freeze}(\mathsf{melt}\, u))$ whenever $\langle\rangle \notin u$. So we get

$$\begin{aligned}
\Phi\big(\mathsf{freeze}(s) * (t * x)\big) &= \Phi\big(\mathsf{freeze}(\mathsf{melt}(\mathsf{freeze}(s) * (t * x)))\big) \\
&= \Phi\big(\mathsf{freeze}(\mathsf{melt}(\mathsf{freeze}(s * t * x)))\big) \\
&= \Phi\big(\mathsf{freeze}(s * t * x)\big).
\end{aligned} \tag{32}$$

Finally, define $\Psi Y H s := \Phi Y H(\mathsf{freeze}\, s)$. Then:

$$\begin{aligned}
\Psi(s) &= \Phi(\mathsf{freeze}\, s) \\
&\overset{(31)}{=} Y\big(\mathsf{melt}(\mathsf{freeze}\, s) * H(\mathsf{melt}(\mathsf{freeze}\, s), \lambda x^\rho \lambda t.\Phi(\mathsf{freeze}(s) * (t * x)))\big) \\
&\overset{(32)}{=} Y\big(s * H(s, \lambda x^\rho \lambda t.\Phi(\mathsf{freeze}(s * t * x)))\big) \\
&= Y(s * H(s, \lambda x^\rho \lambda t.\Psi(s * t * x))),
\end{aligned}$$

which is the desired conclusion. □

5. Fan Functional

The fan functional computes the least modulus of uniform continuity of functionals of the Cantor space into the naturals, where the Cantor space are all 0-1 sequences, i.e., all sequences with values in $\{0, 1\}$ which we encode as sequences of naturals bounded by the constant 1 function.

Let us fix the following notations for the rest of this section: α and β are variables of type $\mathbf{N}^\omega$, p and Y are variables of type $\mathbf{N}^\omega \to \mathbf{N}$, and s is a variable of type $\mathbf{N}^*$. We write $\alpha \leq 1$ for $\forall n\, \alpha(n) \leq 1$, and $s \leq 1$ for $\forall i < |s|\, s_i \leq 1$. Using these notations the fan functional is given by

$$\mathrm{FAN}(\Phi) \quad \Phi(Y) = \min_{n \geq 0}\big(\forall \alpha, \beta \leq 1(\bar{\alpha} n = \bar{\beta} n \to Y\alpha = Y\beta)\big).$$

In fact, the fan functional is a prominent example of a functional which is total (in the sense that FAN is valid in $\mathbb{G}$) but it is *not* computable w.r.t. Kleene's schemes S1–S9 over the Kleene-Kreisel total functionals. The latter was first shown by Tait (unpublished), where he also showed that the fan functional has a recursive associate.

We show that the fan functional is definable by modified bar recursion and Kohlenbach's variant of bar recursion (in $\mathrm{BI}'_D + \mathrm{Cont}$) and thus provide an algorithm for it. The algorithm is divided into two steps. First, we define a search functional using MBR enabling us to decide predicates over the Cantor space. Second, we define a minimization functional using KBR to get the desired witness for the modulus of uniform continuity.

Our algorithm for the fan functional is based on [**6**], which in turn is based on [**3**]. The correctness proofs, however, are our own and we have tried to keep the search functional as general as possible.

5.1. The Search Functional ε. The search functional ε computes witnesses for predicates over the Cantor space whenever they exist. It is given by the following equation

$$\varepsilon ps =_{\mathbf{N}^\omega} s \mathbin{@} [\mathbf{if}\, p(\varepsilon p(s*0))\,\mathbf{then}\,\varepsilon p(s*0)\,\mathbf{else}\,\varepsilon p(s*1)] \tag{33}$$

where $p\colon \mathbf{N}^\omega \to \mathbf{N}$ and $s\colon \mathbf{N}^*$.

Later we shall see that ε has the following property (assuming the continuity of p):

$$\exists \alpha \leq 1\, p(\alpha) = 0 \leftrightarrow p(\varepsilon p\langle\rangle) = 0.$$

First, we show that ε is definable from MBR. Our proof is based on the proof of Theorem 5 in [**6**].

THEOREM 5.1. *ε is definable from* MBR *in* $\mathrm{BI}'_D + \mathrm{Cont}$.

PROOF. We have seen in Lemma 4.17 that the following functional Φ is definable from MBR in $\mathrm{BI}'_D + \mathrm{Cont}$:

$$\Phi(p^{\mathbf{N}^\omega\to\mathbf{N}}, s) =_{\mathbf{N}^\omega} s \mathbin{@} H\big(s, p, \lambda t^{\mathbf{N}^*}\lambda x^{\mathbf{N}}.p(\Phi(p, s*t*x))\big).$$

H is defined by course-of-values recursion as follows

$$H(s,p,F,n) :=_{\mathbf{N}} \begin{cases} s_n & \text{if } n < |s|, \\ 0 & \text{if } n \geq |s| \text{ and } p(F(c,0)) = 0, \\ 1 & \text{if } n \geq |s| \text{ and } p(F(c,0)) \neq 0, \end{cases}$$

where $c := \langle H(s,p,F,|s|), \dots, H(s,p,F,n-1)\rangle$. Fix p and abbreviate Φp by Ψ. We show

$$\forall n.\, \Psi(s)(n) = \Psi\big(s * (\Psi s|s|)\big)(n) \tag{34}$$

by induction on $n \dot{-} |s|$. For $n \leq |s|$ this is trivial.

First observe that for each $n \geq |s|$, we obtain by the defining equation for H and Ψ that

$$\Psi(s)(n) = H(s,p,\lambda t\lambda x.p(\Psi(s*t*x)),n) = \begin{cases} 0 & \text{if } p(\Psi(s*c_{s,n}*0)) = 0, \\ 1 & \text{if } p(\Psi(s*c_{s,n}*0)) \neq 0, \end{cases} \tag{35}$$

where $c_{s,n} := \langle \Psi(s)(|s|), \dots, \Psi(s)(n-1)\rangle$.

Now assume that $n > |s|$. By IH we have for all i with $|s| < i < n$:

$$\Psi(s)(i) = \Psi\big(s * \Psi(s)(|s|)\big)(i)$$

and therefore also

$$c_{s,n} = \langle \Psi(s)(|s|)\rangle * c_{s*\Psi(s)(|s|),n}. \tag{36}$$

By (35) and (36) we obtain:

$$\Psi\big(s * \Psi(s)(|s|)\big)(n) = \begin{cases} 0 & \text{if } p\Big(\Psi\big(s * \langle \Psi(s)(|s|)\rangle * c_{s*\Psi(s)(|s|),n} * 0\big)\Big) = 0, \\ 1 & \text{otherwise}, \end{cases}$$
$$= \begin{cases} 0 & \text{if } p\big(\Psi(s * c_{s,n} * 0)\big) = 0, \\ 1 & \text{otherwise}, \end{cases}$$
$$= \Psi(s)(n).$$

This completes the proof of (34).

We now check that $\Psi(s) = \Phi(p, s)$ satisfies Equation (33) for each argument m. For $m < |s|$ this is trivial since both sides are equal to s_m. So assume $m \geq |s|$. If $p(\Psi(s * 0)) = 0$, then by (35) (with $n = |s|$) $\Psi(s)(|s|) = 0$ so by (34) $\Psi(s)(m) = \Psi(s * 0)(m)$ what we had to show. Similarly for $p(\Psi(s * 0)) \neq 0$. □

THEOREM 5.2. *The following is provable in* $\mathsf{E{-}HA}^\omega + \mathrm{Cont}_{\mathbf{N}} + \varepsilon$ *for all* $p \colon \mathbf{N}^\omega \to \mathbf{N}$ *and* $s \colon \mathbf{N}^*$ *with* $s \leq 1$*:*

(1) $\varepsilon ps \leq 1$,
(2) $\forall \alpha \leq 1. p(\alpha) = 0 \to \forall n\, p(\varepsilon p(\bar{\alpha} n)) = 0$,
(3) $\exists \alpha \in s(\alpha \leq 1 \wedge p(\alpha) = 0) \leftrightarrow p(\varepsilon ps) = 0$,
(4) $\exists \alpha \leq 1\, p(\alpha) = 0 \leftrightarrow p(\varepsilon p\langle\rangle) = 0$.

PROOF. The directions from left to right of (3) and (4) immediately follow from (2) taking $|s|$ respectively 0 for n. The other directions follow from (1).

For (1) one first observes, by a case distinction on whether $p(\varepsilon ps) = 0$ or not, that

$$\varepsilon ps = \varepsilon p(s * (\varepsilon ps|s|)) \quad \text{and} \quad \varepsilon ps|s| \leq 1.$$

By induction on n, this easily generalizes to

$$\varepsilon ps = \varepsilon p(s * d_{s,n}) \quad \text{and} \quad d_{s,n} \leq 1,$$

where $d_{s,0} := \langle\rangle$ and $d_{s,n+1} := d_{s,n} * (\varepsilon p(s * d_{s,n})|s * d_{s,n}|)$. Now given $n \colon \mathbf{N}$ we have

$$\varepsilon psn = \varepsilon p(s * d_{s,n \dot{-} |s|})n \leq 1,$$

which concludes the proof of (1).

For (2) suppose there is an $\alpha \leq 1$ with $p(\alpha) = 0$. By continuity of p there exists n such that

$$\forall \beta \in \bar{\alpha} n\, p(\beta) = 0. \tag{37}$$

Therefore also $p(\varepsilon p(\bar{\alpha} m)) = 0$ for all $m \geq n$. We now prove

$$\forall m \leq n\, p(\varepsilon p(\bar{\alpha} m)) = 0$$

by induction on $n - m$. *Case* $n = m$. See above. *Case* $m < n$. By IH we have

$$p(\varepsilon p(\bar{\alpha}(m + 1))) = 0. \tag{38}$$

In the case $p(\varepsilon p(\bar{\alpha} m * 0)) = 0$ we get $\varepsilon p(\bar{\alpha} m) = \varepsilon p(\bar{\alpha} m * 0)$ and hence $p(\varepsilon p(\bar{\alpha} m))$. In the case $p(\varepsilon p(\bar{\alpha} m * 0)) \neq 0$ we get $\varepsilon p(\bar{\alpha} m) = \varepsilon p(\bar{\alpha} m * 1)$. Moreover, by (38), $\alpha(m)$ must be equal to 1, so (38) yields $p(\varepsilon p(\bar{\alpha} m)) = 0$. This completes the proof. □

REMARK 5.3. Theorem 5.2 shows that it is possible to define universal and existential quantification on the Cantor space with ε. Define the functionals $\exists^C, \forall^C \colon (\mathbf{N}^\omega \to \mathbf{N}) \to \mathbf{N}$ by

$$\exists^C(p) = p(\varepsilon p\langle\rangle) \quad \text{and} \quad \forall^C(p) = p(\varepsilon(\neg p)\langle\rangle),$$

where $\neg p(\alpha) = 1 \dot{-} p(\alpha)$ is the negation of p if p is viewed as a predicate. By the last theorem we get (provably in $\mathsf{E{-}HA}^\omega + \mathrm{Cont}_{\mathbf{N}} + \varepsilon$):

$$\exists^C(p) = 0 \leftrightarrow \exists \alpha \leq 1\, p(\alpha) = 0 \quad \text{and} \quad \forall^C(p) = 0 \leftrightarrow \forall \alpha \leq 1\, p(\alpha) = 0.$$

5.2. Minimization. As we have seen in the last subsection it is possible to decide with ε whether a universal quantified predicate holds on the Cantor space, so in particular we can decide for each n:

$$\forall \alpha, \beta \leq 1(\bar{\alpha}n = \bar{\beta}n \to Y(\alpha) = Y(\beta)).$$

Our next task is to find such an n (which can then also be chosen minimal). This will be done using a slight generalization of Kohlenbach's bar recursion variant KBR.

LEMMA 5.4. KBR *is equivalent to the scheme* KBR$'$ *given by:*

$$\mathrm{KBR}'_{\rho,\tau}(\Phi) \quad \Phi Y G H J s =_\tau \begin{cases} G(s) & \textit{if } Y(s @ 0) =_{\mathbf{N}} Y(s @ J(s)), \\ H(s, \lambda x^\rho . \Phi Y G H J(s * x)) & \textit{otherwise,} \end{cases}$$

where $Y \colon \rho^\omega \to \mathbf{N}$ *and* $s \colon \rho^\omega$.

PROOF. This proof is based on the proof of [**6**, Lemma 3], which in turn is based on [**24**, Theorem 3.66]. Trivially, KBR is definable from KBR$'$. For the converse direction, we first introduce some notations. The lifting of addition of naturals to higher types, $x^\rho + n^{\mathbf{N}}$, is defined componentwise by induction on ρ. In the same manner, cut-off subtraction $x^\rho \dot{-} n^{\mathbf{N}}$ is lifted to higher types. Recall the definitions of ψ_ρ and $\mathbf{emb}_\rho$ from the beginning of this chapter. By (meta-) induction on ρ one immediately verifies that

$$\psi_\rho(\mathbf{emb}_\rho\, n) = n, \psi_\rho(x^\rho + 2) > 1, \text{ and } (x^\rho + n) \dot{-} n = x^\rho.$$

Define

$$\eta(J^{\rho^* \to \rho^\omega}, \beta^{\rho^\omega}, n) :=_\rho \begin{cases} \beta(n) \dot{-} 2 & \text{if } \psi_\rho(\beta(n)) > 1, \\ J(\varphi(\bar{\beta}(n+1)) \dot{-} 2, n) & \text{if } \psi_\rho(\beta(n)) = 1, \\ 0^\rho & \text{if } \psi_\rho(\beta(n)) = 0, \end{cases}$$

where $\varphi(s) := \langle s_0, \ldots, s_{k-1} \rangle$ with $k = \min_{n<|s|}(\psi_\rho(s_n) = 1)$. By componentwise comparison, it is easy to see that the following equations hold:

(i) $\eta(J, (s+2) @ 0^{\rho^\omega}) = s @ 0^{\rho^\omega}$,
(ii) $\eta(J, (s+2) @ 1^{\rho^\omega}) = s @ J(s)$.

Using KBR, we can define

$$\Psi'(s) =_\tau \begin{cases} G(s \dot{-} 2) & \text{if } Y(\eta(J, s @ 0^{\rho^\omega})) =_{\mathbf{N}} Y(\eta(J, s @ 1^{\rho^\omega})), \\ H\big(s \dot{-} 2, \lambda x^\rho . \Psi'(s * (x^\rho + 2))\big) & \text{otherwise,} \end{cases}$$

where Ψ' is short for $\Psi Y G H J$. We leave it to the reader to verify that $\Phi Y G H J s := \Psi Y G H J(s+2)$ satisfies KBR$'$. ☐

Now we have all tools at hand to define the fan functional. We argue informally in $\mathsf{E\text{-}HA}^\omega + \mathrm{Cont}_{\mathbf{N}} + \varepsilon + \mathrm{KBR}$. First we use the search functional ε to define Φ as

$$\Phi(Y, s, m) := \varepsilon(\lambda\alpha.\alpha \in s \wedge Y(\alpha) \neq m)s. \tag{39}$$

Note that $\Phi(Y, s, m) = s \mathbin{@} \Phi(Y, s, m)$ so using KBR and Lemma 5.4 we can define

$$\Psi(Y, s) :=_{\mathbf{N}} \begin{cases} 0 & \text{if } Y(\Phi(Y, s, Y(s \mathbin{@} 0))) = Y(s \mathbin{@} 0), \\ 1 + \max\{\Psi(Y, s * 0), \Psi(Y, s * 1)\} & \text{otherwise.} \end{cases} \tag{40}$$

Theorem 5.2 yields for all m and s with $s \leq 1$:

$$\exists\alpha \leq 1(\alpha \in s \wedge Y(\alpha) \neq m) \leftrightarrow Y(s \mathbin{@} \Phi(Y, s, m)) \neq m.$$

In particular, we obtain for $m = Y(s \mathbin{@} 0)$:

$$\forall\alpha \leq 1\big(\alpha \in s \to Y(\alpha) = Y(s \mathbin{@} 0)\big) \leftrightarrow Y\big(s \mathbin{@} \Phi(Y, s, Y(s \mathbin{@} 0))\big) = Y(s \mathbin{@} 0),$$

which by definition of Ψ gives

$$\forall\alpha \leq 1\big(\alpha \in s \to Y(\alpha) = Y(s \mathbin{@} 0)\big) \leftrightarrow \Psi(Y, s) = 0. \tag{41}$$

Lemma 5.5. *For all $Y, s \leq 1$, and $\alpha \leq 1$ we have*

$$\alpha \in s \to Y(\alpha) = Y(\overline{\alpha, |s| + \Psi(Y, s)}).$$

Proof. $\mathrm{Ind}(\Psi(Y, s))$. *Case* $\Psi(Y, s) = 0$. Use (41). *Case* $\Psi(Y, s) > 0$. Let $\alpha \in s$. We have to show:

$$Y(\alpha) = Y(\overline{\alpha, |s| + \Psi(Y, s)}).$$

By the definition of Ψ, $\Psi(Y, s) > \Psi(Y, s * \alpha|s|)$. So by IH

$$\forall\beta \leq 1\Big(\beta \in s * \alpha|s| \to Y(\beta) = Y\big(\overline{\beta, |s| + 1 + \Psi(Y, s * \alpha|s|)}\big)\Big).$$

In particular, we obtain (since $\alpha, (\overline{\alpha, |s| + \Psi(Y, s)}) \in s * \alpha|s|$)

(i) $Y(\alpha) = Y(\overline{\alpha, |s| + 1 + \Psi(Y, s * \alpha|s|)})$, and

(ii) $Y(\overline{\alpha, |s| + \Psi(Y, s)}) = Y\Big(\overline{(\overline{\alpha, |s| + \Psi(Y, s)}), |s| + 1 + \Psi(Y, s * \alpha|s|)}\Big)$.

Moreover:

$$|s| + 1 + \Psi(Y, s * \alpha|s|) \leq |s| + 1 + \max_{i=0,1} \Psi(Y, s * i) = |s| + \Psi(Y, s).$$

Using this inequality we get

$$\Big(\overline{(\overline{\alpha, |s| + \Psi(Y, s)}), |s| + 1 + \Psi(Y, s * \alpha|s|)}\Big) = (\overline{\alpha, |s| + 1 + \Psi(Y, s * \alpha|s|)}).$$

So combining this, (i), and (ii) gives:

$$\begin{aligned} Y(\alpha) &\overset{\text{(i)}}{=} Y(\overline{\alpha, |s| + 1 + \Psi(Y, s * \alpha|s|)}) \\ &= Y\Big(\overline{(\overline{\alpha, |s| + \Psi(Y, s)}), |s| + 1 + \Psi(Y, s * \alpha|s|)}\Big) \\ &\overset{\text{(ii)}}{=} Y(\overline{\alpha, |s| + \Psi(Y, s)}). \end{aligned}$$

This is what we had to prove. □

Lemma 5.6. *$\Psi(Y, s)$ is minimal w.r.t. the property of Lemma 5.5.*

PROOF. Ind($\Psi(Y,s)$). *Case* $\Psi(Y,s)=0$. Trivial. *Case* $\Psi(Y,s)>0$. Let n be given with:

$$\forall \alpha \le 1\bigl(\alpha \in s \to Y(\alpha) = Y(\overline{\alpha, |s|+n})\bigr). \tag{42}$$

Then $n \neq 0$ by (41). Assume that

$$n < \Psi(Y,s) = 1 + \max_{i=0,1} \Psi(Y, s*i). \tag{43}$$

By (43) there exists $i \in \{0,1\}$ such that $n < 1 + \Psi(Y, s*i)$ and by (42) we get

$$\forall \alpha \le 1\bigl(\alpha \in s*i \to Y(\alpha) = Y(\overline{\alpha, |s|+n})\bigr).$$

But we also have

$$|s*i| + (n-1) = |s| + n < |s| + 1 + \Psi(Y, s*i) = |s*i| + \Psi(Y, s*i),$$

which is a contradiction to the IH for $s*i$, and hence $n \ge \Psi(Y,s)$ as required. □

With Lemma 5.5 we get $Y(\alpha) = Y(\overline{\alpha, \Psi(Y, \langle\rangle)})$ for all $\alpha \le 1$, so in particular:

$$\forall \alpha, \beta \le 1\bigl(\bar\alpha(\Psi(Y,\langle\rangle)) = \bar\beta(\Psi(Y,\langle\rangle)) \to Y(\alpha) = Y(\beta)\bigr).$$

Lemma 5.6 implies that $\Psi(Y,\langle\rangle)$ is minimal with this property, so in other words $\lambda Y.\Psi Y\langle\rangle$ is the fan functional.

THEOREM 5.7. *The fan functional is definable from* ε + KBR *in* $\mathrm{Cont}_{\mathbf{N}}$. *In particular, it is definable from* MBR + KBR *in* BI'_D + Cont.

REMARK 5.8. Both ε and Ψ from above are definable by the fan functional. For ε note that εp is primitive recursively definable in the modulus of uniform continuity of p. Using Lemma 5.5 and 5.6 it is also not hard to see that $\Psi(Y,s)$ is equal to the fan functional at $\lambda\alpha.Y(s \mathbin{@} \alpha)$.

REMARK 5.9. Instead of looking at the Cantor space, i.e., all sequences bounded by one, one can also look, more generally, at the sequences bounded by a fixed sequence. Our treatment above can be generalized to this setting: Similarly as above one can define a general form of the fan functional Φ' satisfying

$$\Phi'(Y,\gamma) = \min_{n\ge 0}\bigl(\forall \alpha, \beta \le \gamma(\bar\alpha n = \bar\beta n \to Y\alpha = Y\beta)\bigr),$$

where $\alpha \le \gamma$ is defined componentwise by $\forall n\, \alpha(n) \le \gamma(n)$.

6. Notes

We conclude this chapter with some remarks on definability and another variant of bar recursion.

In [**8**], Bezem introduced his model $\mathfrak{M}$ of strongly majorizable functionals and showed that it is a model of Spector's bar recursion. The novelty of the model $\mathfrak{M}$ was that it contains *discontinuous* functionals — even one at type $(\mathbf{N} \to \mathbf{N}) \to \mathbf{N}$ which shows that the fan functional does not exist in $\mathfrak{M}$, in particular, it is not definable by SBR. Berger and Oliva [**6, 7**] showed that modified bar recursion exists in $\mathfrak{M}$ and thus the fan functional is not definable by MBR alone either. Moreover, they showed that MBR is not S1–S9 computable over the Kleene-Kreisel continuous functionals; on the other hand, KBR (as well as SBR) is S1–S9 computable over the Kleene-Kreisel continuous functionals but does not exist in $\mathfrak{M}$ (cf. [**24**]), and thus neither MBR defines KBR nor conversely. As already stated in the last section, the fan functional is not S1–S9 computable over the Kleene-Kreisel functionals and thus it is not definable by KBR alone. In [**24**, p. 57 ff.], it was also shown that KBR defines SBR.

There are various open problems concerning interdefinability. We have seen that OR defines EUR which in turn defines UR. What about the converse directions? What is the exact relation between BBC, OR/EUR/UR, and MBR? Does the wBBC functional define the BBC functional?

Recently, Escardó and Oliva [**15**] (inspired by [**13**]) have proposed another variant of bar recursion called *course-of-values bar recursion*, which can be written as

$$\Psi Y H s =_{\rho^\omega} s \mathbin{@} \lambda n. H\big(s, n, \lambda x^\rho. Y\big(\Psi Y H((\overline{\Psi Y H s})(n) * x)\big)\big).$$

In [**14**], it has been reported that this scheme is equivalent to modified bar recursion.

CHAPTER 4

Proof Interpretations

This chapter is devoted to the computational content of proofs. Starting from proofs in classical analysis (i.e., Peano Arithmetic with countable choice or dependent choice), we extract an algorithm together with a proof that the algorithm satisfies a specification depending on the formula we have proven. Here our main techniques are negative translation, A-translation, and modified realizability.

NOTATION. In this chapter we work in HA^ω (with base types as in Chapter 3), i.e., we assume that the set $\mathcal{X}$ of predicate variables is empty. By formula we always mean a HA^ω-formula; the same for predicates. A term is always a term of HA^ω. All axiom schemes (e.g., AC, DC, OI, EUI, UI) are restricted to the language of HA^ω.

It is convenient to allow arbitrary return types of level zero for open-, update-, and extended update recursion. For example, for open recursion this means that our defining equation becomes

$$\mathrm{OR}_{\prec,\rho,\tau}(\mathcal{R}^{\circ}_{\prec}) \quad \mathcal{R}^{\circ}_{\prec}F\alpha =_\tau F\alpha\big(\lambda n, x^\rho, \beta.\mathbf{if}\ x \prec \alpha n\ \mathbf{then}\ \mathcal{R}^{\circ}_{\prec}F(\bar{\alpha}n * x \mathbin{@} \beta)\big)$$

where τ is a type with $\mathrm{lev}(\tau) = 0$ and $\prec$ is primitive recursive with $\mathsf{HA}^\omega \vdash \mathrm{TI}_\prec$.

This poses no restriction because each type of level zero is primitive recursively isomorphic to $\mathbf{N}$, and thus it is not hard to see that the more general scheme with return types of level zero are equivalent (in the sense of Definition 4.1 in Chapter 3) to the respective scheme with only $\mathbf{N}$ as a return type.

1. Negative Translation and A-Translation

As a first step, we will transform a proof in classical logic into one in minimal logic. For this we use the well-known negative translation of Gödel and Gentzen.

DEFINITION 1.1. The *negative translation* A^{g} of a formula A is defined by induction on A:

$$\begin{aligned} P^{\mathrm{g}} &:= \neg\neg P \text{ if } P \text{ is atomic,} \\ (A \circ B)^{\mathrm{g}} &:= A^{\mathrm{g}} \circ B^{\mathrm{g}} \text{ for } \circ = \wedge, \to, \\ (A \vee B)^{\mathrm{g}} &:= \neg(\neg A^{\mathrm{g}} \wedge \neg B^{\mathrm{g}}), \\ (\forall x A)^{\mathrm{g}} &:= \forall x A^{\mathrm{g}}, \\ (\exists x A)^{\mathrm{g}} &:= \neg\forall x \neg A^{\mathrm{g}}. \end{aligned}$$

LEMMA 1.2. *(1)* $\vdash_{\mathrm{m}} \neg\neg A^{\mathrm{g}} \to A^{\mathrm{g}}$

(2) If $\Gamma \vdash_{\mathrm{c}} A$*, then* $\Gamma^{\mathrm{g}} \vdash_{\mathrm{m}} A^{\mathrm{g}}$.

PROOF. (1) is proved by induction on A and (2) by induction on the derivation $\Gamma \vdash_{\mathrm{c}} A$ using (1) for stability. □

A-translation is an elegant and flexible method to show Π^0_2-conservativity of various classical formal systems over their intuitionistic counterparts. It was independently introduced by Friedman [**16**] and Dragalin [**11**].

DEFINITION 1.3. Let A be a formula. We define the *A-translation* B_A of the formula B by induction on B:

$$\begin{aligned} \bot_A &:= A, \\ P_A &:= P \vee A \text{ if } P \text{ is atomic } \neq \bot, \\ (B \circ C)_A &:= B_A \circ C_A \text{ for } \circ = \wedge, \vee, \to, \\ (\mathsf{Q}xB)_A &:= \mathsf{Q}xB_A \text{ for } \mathsf{Q} = \forall, \exists, \end{aligned}$$

where in the quantifier case, we assume that x is not free in A.

LEMMA 1.4. *(1)* $\vdash_{\mathrm{m}} A \to B_A$
(2) If $\Gamma \vdash_{\mathrm{i}} B$*, then* $\Gamma_A \vdash_{\mathrm{m}} B_A$.

PROOF. (1) is proved by induction on B, or by observing that A occurs strictly positive in B_A, and (2) is proved by induction on the derivation $\Gamma \vdash_{\mathrm{i}} B$ using (1) for ex falso quodlibet. □

LEMMA 1.5. *Let B be a $\to\forall$-free formula. Then:*

(1) $\vdash_{\mathrm{m}} B^{\mathrm{g}} \leftrightarrow \neg\neg B$,
(2) $\vdash_{\mathrm{i}} B_A \leftrightarrow B \vee A$.

PROOF. Induction on B. □

LEMMA 1.6. *Let A be a $\to\forall$-free formula.*

(1) If $\Gamma \vdash_{\mathrm{i}} \Gamma_A$ *and* $\Gamma \vdash_{\mathrm{i}} \neg\neg A$*, then* $\Gamma \vdash_{\mathrm{i}} A$.
(2) If $\Gamma \vdash_{\mathrm{i}} \Gamma^{\mathrm{g}}$ *and* $\Gamma \vdash_{\mathrm{i}} \Gamma_A$*, then* $\Gamma \vdash_{\mathrm{c}} A$ *implies* $\Gamma \vdash_{\mathrm{i}} A$.

PROOF. For (1) assume $\Gamma \vdash_{\mathrm{i}} \Gamma_A$ and $\Gamma \vdash_{\mathrm{i}} \neg\neg A$. Then by Lemma 1.4 (2) we conclude $\Gamma_A \vdash_{\mathrm{m}} (A_A \to A) \to A$, and thus $\Gamma \vdash_{\mathrm{i}} (A_A \to A) \to A$. But by Lemma 1.5 (2) $\vdash_{\mathrm{i}} A_A \to A$, so we conclude $\Gamma \vdash_{\mathrm{i}} A$.

For (2) suppose $\Gamma \vdash_{\mathrm{i}} \Gamma^{\mathrm{g}} \cup \Gamma_A$ and $\Gamma \vdash_{\mathrm{c}} A$. Then Lemma 1.2 (2) gives us $\Gamma^{\mathrm{g}} \vdash_{\mathrm{m}} A^{\mathrm{g}}$, and thus $\Gamma \vdash_{\mathrm{i}} A^{\mathrm{g}}$ by our assumption. Lemma 1.5 (1) yields $\Gamma \vdash_{\mathrm{i}} \neg\neg A$, so the claim follows from (1). □

DEFINITION 1.7. Let Δ and Γ be sets of formulas. Then Γ is *closed under negative translation* if $\Gamma \vdash_{\mathrm{i}} \Gamma^{\mathrm{g}}$. Γ is *closed under Δ-translation* if $\Gamma \vdash_{\mathrm{i}} \Gamma_A$ for all $A \in \Delta$.

COROLLARY 1.8. *Let Δ be a set of $\to\forall$-free formulas and Γ be closed under negative- and Δ-translation. Then* $\Gamma \vdash_{\mathrm{c}} A$ *implies* $\Gamma \vdash_{\mathrm{i}} A$ *for all* $A \in \Delta$.

A Digression: Barr's Theorem. As a further application of Lemma 1.6 we prove Barr's Theorem following Palmgren [**32**]. A $\to\forall$-free formula A is called *geometric*. A *geometric implication* is a formula of the form $\forall\vec{x}(A \to B)$ with A and B geometric. Finally, a *geometric theory* is a set of formulas which are either geometric implications or geometric formulas.

COROLLARY 1.9 (Barr's Theorem). *Let Γ be a geometric theory and A be a geometric implication. Then $\Gamma \vdash_{\mathrm{c}} A$ implies $\Gamma \vdash_{\mathrm{i}} A$.*

PROOF. By instantiating the outer universal quantifiers with fresh variables and moving the premise of A to Γ, we may assume that A is geometric. By Lemma 1.6, it suffices to show that Γ is closed under negative- and A-translation. For geometric implications in Γ the former easily follows from $\vdash_{\mathrm{m}} (B \to C) \to \neg\neg B \to \neg\neg C$ together with Lemma 1.5 (1). For the latter, use $\vdash_{\mathrm{m}} (B \to C) \to B \vee A \to C \vee A$ together with Lemma 1.5 (2). Similarly for the geometric formulas in Γ. $\square$

2. Realizability Interpretation

The modified realizability interpretation was introduced by Kreisel in [**26**]. It provides a method to extract the computational content of proofs in various systems of intuitionistic arithmetic. We proceed as follows. To each formula A we assign a type (or a dummy symbol ε indicating no content) $\tau(A)$ for the type of its potential realizers. Then we define the formula $t\,\mathbf{mr}\,A$ where t is a term (or a dummy symbol if $\tau(A) = \varepsilon$) which specifies what it means that t is the computational content of A. The Soundness Theorem guarantees that from a proof of A we can in fact obtain a term t and a proof of $t\,\mathbf{mr}\,A$.

DEFINITION 2.1. The *type* $\tau(A)$ of a formula A is either a type or a special symbol (called the nulltype) denoted by ε. Type constructors are extended to the nulltype symbol as follows, for ρ either a type or ε:

$$\varepsilon \times \rho := \rho \times \varepsilon := \varepsilon \to \rho := \rho,$$
$$\varepsilon^* := \rho \to \varepsilon := \varepsilon.$$

With these conventions, $\tau(A)$ is defined by induction on A:

$$\begin{aligned}
\tau(A_0) &:= \varepsilon \quad \text{if } A_0 \text{ is atomic},\\
\tau(A \wedge B) &:= \tau(A) \times \tau(B),\\
\tau(A \to B) &:= \tau(A) \to \tau(B),\\
\tau(\forall x^\rho A) &:= \rho \to \tau(A),\\
\tau(A \vee B) &:= \mathbf{N} \times (\tau(A) \times \tau(B)),\\
\tau(\exists x^\rho A) &:= \rho \times \tau(A).
\end{aligned}$$

DEFINITION 2.2. Let A be a formula and t a term of type $\tau(A)$ if $\tau(A)$ is a type and otherwise a special nullterm symbol ϵ. We extend application to the nullterm symbol for s a term or ϵ by $\epsilon s := \epsilon$ and $s\epsilon := s$. The formula $t\,\mathbf{mr}\,A$ (read as t *realizes* A, or t is a *realizer* of A) is defined by induction on A:

$$\begin{aligned}
\epsilon \,\mathbf{mr}\, A_0 &:= A_0 \quad \text{if } A_0 \text{ is atomic},\\
t \,\mathbf{mr}\, (A \wedge B) &:= \pi_0(t) \,\mathbf{mr}\, A \wedge \pi_1(t) \,\mathbf{mr}\, B,\\
t \,\mathbf{mr}\, (A \to B) &:= \forall x^{\tau(A)} (x \,\mathbf{mr}\, A \to tx \,\mathbf{mr}\, B),\\
t \,\mathbf{mr}\, \forall x^\rho A &:= \forall x^\rho\, tx \,\mathbf{mr}\, A,\\
t \,\mathbf{mr}\, (A \vee B) &:= (\pi_0(t) = 0 \to \pi_0(\pi_1(t)) \,\mathbf{mr}\, A) \wedge\\
&\qquad (\pi_0(t) \neq 0 \to \pi_1(\pi_1(t)) \,\mathbf{mr}\, B),\\
t \,\mathbf{mr}\, \exists x^\rho A(x) &:= \pi_1(t) \,\mathbf{mr}\, A(\pi_0(t)),
\end{aligned}$$

where in the quantifier cases, we assume $x \notin \mathsf{FV}(t)$. In the $\wedge$-case the following conventions apply. If $\tau(A) = \varepsilon$, we set $\pi_0(t) := \epsilon$ and $\pi_1(t) := t$; if $\tau(B) = \varepsilon$, we set $\pi_0(t) := t$ and $\pi_1(t) := \epsilon$. Similar conventions for projections apply in the $\vee$ and $\exists$ cases. Moreover, the expression $\forall x^\varepsilon C(x)$ should be read as $C(\epsilon)$.

Note that for any formula A, $t \,\mathbf{mr}\, A$ contains neither existential quantifiers nor disjunctions.

THEOREM 2.3 (Soundness). *Let M be a minimal logic derivation in HA^ω of A from assumptions $u_1 \colon C_1, \dots, u_n \colon C_n$.*[1] *Then there exists a term $[\![M]\!]$ and a minimal logic derivation M' in HA^ω of $[\![M]\!] \,\mathbf{mr}\, A$ from assumptions $\bar{u}_1 \colon x_1 \,\mathbf{mr}\, C_1, \dots, \bar{u}_n \colon x_n \,\mathbf{mr}\, C_n$. Moreover, the free variables of $[\![M]\!]$ are among $x_1, \dots, x_n$ and the free variables of terms in the $\forall$-eliminations and $\exists$-introductions of M.*

PROOF. See, e.g., [**35**] for a proof in our setting. The claim on the free variables is easily obtained by inspecting the proof given there. □

3. Realizing Open Induction and Variants

In the present section we show that open recursion realizes open induction. This provides us with a direct method for extracting programs from classical proofs using open induction. It turns out that realizing (extended) update induction by (extended) update recursion is not directly possible, but if we restrict (extended) update induction to certain pointwise open predicates, this is realizable by (extended) update recursion. The restriction of (extended) update induction to pointwise open predicates is still strong enough to prove AC^{g}, which results in an extraction method for classical proofs using AC by (extended) update recursion.

This section is mainly based on [**4**], except that the treatment of extended update induction is our own (but along the lines of the treatment of update induction given loc. cit.) and we have filled a gap in Theorem 4.4 loc. cit. (cf. Remark 3.7).

Let Σ be the set of all Σ-formulas.

LEMMA 3.1. *HA^ω as well as HA^ω augmented by each of open induction, update induction, and extended update induction are closed under negative- and Σ-translation.*

[1] The u's are assumption variables.

PROOF. It is not hard to see that HA^ω is closed under negative- and Σ-translation.

Let $U(\alpha) = C^\forall(\alpha) \to B^\exists(\alpha)$ be open with C arbitrary and $B \in \Sigma$. Fix $A \in \Sigma$. Clearly, the A-translation U_A of an open predicate U is open again. We now calculate $(\mathrm{OI}_{\prec,U})_A$:

$$\Big(\forall\alpha\big(\forall n, x^\rho, \gamma(x \prec \alpha n \to U(\bar{\alpha}n * x \mathbin{@} \gamma)) \to U(\alpha)\big) \to \forall\alpha U(\alpha)\Big)_A$$
$$= \forall\alpha\big(\forall n, x^\rho, \gamma(x \prec \alpha n \vee A \to U_A(\bar{\alpha}n * x \mathbin{@} \gamma)) \to U_A(\alpha)\big) \to \forall\alpha U_A(\alpha).$$

By Lemma 1.4 (1), we get

$$\big(x \prec \alpha n \vee A \to U_A(\bar{\alpha}n * x \mathbin{@} \gamma)\big) \leftrightarrow \big(x \prec \alpha n \to U_A(\bar{\alpha}n * x \mathbin{@} \gamma)\big)$$

and thus $\mathrm{OI}_{\prec,U_A}$ implies $(\mathrm{OI}_{\prec,U})_A$.

Concerning the negative translation, we have

$$\begin{aligned} U^{\mathrm{g}}(\alpha) &= \big(C^\forall(\alpha) \to B^\exists(\alpha)\big)^{\mathrm{g}} \\ &= \forall n C^{\mathrm{g}}(\bar{\alpha}n) \to \neg\forall n \neg B^{\mathrm{g}}(\bar{\alpha}n) \\ &\leftrightarrow (C^g \wedge \neg B^{\mathrm{g}})^\forall(\alpha) \to \bot. \end{aligned}$$

And hence we may assume that U^{g} is open again. Now it is easy to see that $\mathrm{OI}_{\prec,U^{\mathrm{g}}}$ implies $\big(\mathrm{OI}_{\prec,U}\big)^{\mathrm{g}}$ using the stability of $\prec$ (i.e., $\neg\neg x \prec y \to x \prec y$) which is derivable in HA^ω because $\prec$ is primitive recursive.

Similarly for update and extended update induction. □

COROLLARY 3.2. *Let* $\Box \in \{\mathrm{OI}, \mathrm{EUI}, \mathrm{UI}\}$ *and* A *be a* Σ*-formula. Then:*

$$\mathsf{PA}^\omega + \Box \vdash_{\mathrm{c}} \forall\vec{x}A \Rightarrow \mathsf{HA}^\omega + \Box \vdash_{\mathrm{i}} \forall\vec{x}A.$$

PROOF. Immediate from Corollary 1.8 and the last lemma.[2] □

LEMMA 3.3. *Let* A *be a* Σ*-formula. Then we have*

$$\mathsf{HA}^\omega + \mathrm{Cont} \vdash_{\mathrm{m}} \forall\alpha\big(A(\alpha) \to \exists n \forall\beta(\bar{\alpha}n = \bar{\beta}n \to A(\beta))\big).$$

PROOF. Induction on A. In all cases, except when A is atomic, the IH suffices. If A is atomic, proceed as follows. First, using primitive recursive isomorphism between $\mathbf{N}$ and any type of level zero, one generalizes Cont to return types τ with $\mathrm{lev}(\tau) = 0$:

$$\mathsf{HA}^\omega + \mathrm{Cont} \vdash_{\mathrm{m}} \forall F^{\rho^\omega \to \tau} \forall\alpha\exists n \forall\beta \in \bar{\alpha}n(F(\alpha) =_\tau F(\beta)).$$

Recall that by the definition of Σ-formula, A is of the form $P(\vec{t})$ where P is a predicate of arity $(\vec{\sigma})$ and $\mathrm{lev}(\sigma) = 0$ for each $\sigma \in \vec{\sigma}$. Combined with the generalization of continuity, this easily yields the claim. □

LEMMA 3.4. *Let* B *be a* Σ*-formula.*

(i) $x \,\mathbf{mr}\, B$ *is equivalent (over* HA^ω*) to a* Σ*-formula.*
(ii) $\mathsf{HA}^\omega \vdash_{\mathrm{m}} B \leftrightarrow \exists x\, x \,\mathbf{mr}\, B$.

PROOF. Both claims are proved by induction on B, where for (i) one uses:

$$\begin{aligned} x \,\mathbf{mr}\, (A \vee B) \leftrightarrow\ & (\pi_0(x) = 0 \wedge \pi_0(\pi_1(x)) \,\mathbf{mr}\, A) \vee \\ & (\pi_0(x) \neq 0 \wedge \pi_1(\pi_1(x)) \,\mathbf{mr}\, B). \end{aligned}$$ □

[2] Observe that the involved translations do not alter the variable condition for $\forall$-introductions.

REMARK 3.5. Let $U(\alpha^{\rho^\omega}) = C^\forall(\alpha) \to B^\exists(\alpha)$ be open, $\sigma := \tau(C)$, and $\tau := \tau(B)$. We identify a pair of sequences $\langle \alpha, \delta \rangle \colon \rho^\omega \times \sigma^\omega$ with the sequence $\lambda n.\langle \alpha n, \delta n \rangle \colon (\rho \times \sigma)^\omega$ of pairs. In the presence of extensionality or η-equality, this identification is unproblematic. Let t be a term of type $\rho^\omega \to \sigma^\omega \to \mathbf{N} \times \tau$. Then, in $\mathsf{E{-}PA}^\omega$, the predicate

$$V(\alpha, \delta) := \delta \,\mathbf{mr}\, C^\forall(\alpha) \to t(\alpha, \delta) \,\mathbf{mr}\, B^\exists(\alpha)$$

is equivalent to an open (w.r.t. $(\rho \times \sigma)^\omega$) predicate. Hence, we may assume that $V(\alpha, \delta)$ is open.

PROOF. We have $\delta \,\mathbf{mr}\, C^\forall(\alpha) = \forall n(\delta n \,\mathbf{mr}\, C(\bar{\alpha} n))$; it is not hard to see that this is equivalent to $D^\forall(\alpha, \delta)$, where

$$D(s^{\rho^*}, u^{\sigma^*}) :\leftrightarrow s = \langle\rangle \vee \big(s \neq \langle\rangle \wedge u \neq \langle\rangle \wedge \mathsf{last}(u) \,\mathbf{mr}\, C(\mathsf{tail}(s))\big),$$

last is defined such that $\mathsf{last}(u * x^\sigma) = x$, and tail is such that $\mathsf{tail}(s * x^\sigma) = s$. Hence the premise is already of the form we need.

By Lemma 3.4 we may assume that

$$t(\alpha, \delta) \,\mathbf{mr}\, B^\exists(\alpha)$$

is a Σ-formula. Note that $\mathrm{lev}(\tau) = 0$ because B is a Σ-formula (cf. the definition on p. 54). Lemma 3.3 implies for all α and δ:

$$t(\alpha, \delta) \,\mathbf{mr}\, B^\exists(\alpha) \leftrightarrow \exists n \, W(\bar{\alpha} n, \bar{\delta} n) = W^\exists(\alpha, \delta),$$

where

$$W(u^{\rho^*}, v^{\sigma^*}) := \forall \beta, \varepsilon(\beta \in u \wedge \varepsilon \in v \to t(\beta, \varepsilon) \,\mathbf{mr}\, B^\exists(\beta)).$$

Note that W is not $\to \forall$-free, but using classical logic we can apply Remark 3.3 from Chapter 3 to get the desired result. ☐

THEOREM 3.6. *There exists a closed term t such that*

$$\mathsf{E{-}PA}^\omega + \mathrm{OI} + \mathrm{Cont} + \mathrm{OR}(\mathcal{R}^\circ) \vdash_{\mathrm{c}} t(\mathcal{R}^\circ) \,\mathbf{mr}\, \mathrm{OI}.$$

PROOF. Let $U(\alpha^{\rho^\omega}) = C^\forall(\alpha) \to B^\exists(\alpha)$ be open where B is a Σ-formula and let $\prec$ be a primitive recursive relation of arity (ρ, ρ) with $\mathsf{HA} \vdash_{\mathrm{i}} \mathrm{TI}_\prec$. Set $\tau := \tau(B)$ and $\sigma := \tau(C)$. Observe that $\mathrm{lev}(\tau) = 0$ because $B \in \Sigma$. Define $\prec'$ by $\langle x_1, y_1 \rangle^{\rho \times \sigma} \prec' \langle x_2, y_2 \rangle^{\rho \times \sigma} :\leftrightarrow x_1 \prec x_2$. Using open recursion for $\prec'$ we can define a functional Ψ satisfying

$$\begin{aligned} &\Psi F \alpha^{\rho^\omega} \delta^{\sigma^\omega} =_{\mathbf{N} \times \tau} \\ &\quad F\alpha\big(\lambda n, x^\rho, \gamma^{\rho^\omega}, \eta^{\sigma^\omega}. \mathbf{if}\ x \prec \alpha n \ \mathbf{then}\ \Psi F(\bar{\alpha} n * x @ \gamma)(\bar{\delta} n @ \eta)\big)\delta. \end{aligned} \tag{44}$$

We claim that Ψ realizes open induction. Let F be a realizer of the premise, i.e.,

$$F \,\mathbf{mr}\, \forall \alpha \left[\forall n, x^\rho, \gamma^{\rho^\omega}(x \prec \alpha n \to U(\bar{\alpha} n * x @ \gamma)) \to U(\alpha)\right]. \tag{45}$$

Let $V(\alpha, \delta) := \delta \,\mathbf{mr}\, C^\forall(\alpha) \to \Psi F \alpha \delta \,\mathbf{mr}\, B^\exists(\alpha)$. We have to verify

$$\forall \alpha, \delta \, V(\alpha, \delta).$$

By Remark 3.5, we may assume that V is open, and thus can argue by open induction. Fix α and δ with

$$\delta \,\mathbf{mr}\, C^\forall(\alpha). \tag{46}$$

We have to prove $\Psi F\alpha\delta \,\mathbf{mr}\, B^{\exists}(\alpha)$. Equation (45) unfolds to

$$\forall\alpha\forall\xi \begin{bmatrix} \forall n, x, \gamma\big(x \prec \alpha n \to \xi n x \gamma \,\mathbf{mr}\, U(\bar{\alpha}n * x \mathbin{@} \gamma)\big) \to \\ \forall\delta\big(\delta \,\mathbf{mr}\, C^{\forall}(\alpha) \to F\alpha\xi\delta \,\mathbf{mr}\, B^{\exists}(\alpha)\big) \end{bmatrix}. \tag{47}$$

So by (44), (47), and (46), it suffices to prove

$$\lambda\eta.\Psi F(\bar{\alpha}n * x \mathbin{@} \gamma)(\bar{\delta}n \mathbin{@} \eta) \,\mathbf{mr}\, U(\bar{\alpha}n * x \mathbin{@} \gamma)$$

for all n, x, and γ with $x \prec \alpha n$. So assume $x \prec \alpha n$ and

$$\eta \,\mathbf{mr}\, C^{\forall}(\bar{\alpha}n * x \mathbin{@} \gamma). \tag{48}$$

We have to show

$$\Psi F(\bar{\alpha}n * x \mathbin{@} \gamma)(\bar{\delta}n \mathbin{@} \eta) \,\mathbf{mr}\, B^{\exists}(\bar{\alpha}n * x \mathbin{@} \gamma).$$

By the open induction hypothesis we have

$$\forall m, y, \varphi, \psi \left(y \prec \alpha m \to V(\bar{\alpha}m * y \mathbin{@} \psi, \bar{\delta}m \mathbin{@} \varphi)\right),$$

in particular, $V(\bar{\alpha}n * x \mathbin{@} \gamma, \bar{\delta}n \mathbin{@} \eta)$. So it suffices to prove:

$$(\bar{\delta}n \mathbin{@} \eta) \,\mathbf{mr}\, C^{\forall}(\bar{\alpha}n * x \mathbin{@} \gamma), \text{ i.e., } \forall k\, (\bar{\delta}n \mathbin{@} \eta)k \,\mathbf{mr}\, C\big((\overline{\bar{\alpha}n * x \mathbin{@} \gamma})k\big).$$

For $k < n$ this amounts to proving $\delta k \,\mathbf{mr}\, C(\bar{\alpha}k)$ which follows from (46). For $k \geq n$ we need $\eta k \,\mathbf{mr}\, C\big((\overline{\bar{\alpha}n * x \mathbin{@} \gamma})k\big)$ which is immediate from (48). This completes the proof. □

Remark 3.7. In [**4**], the last theorem is stated with HA^{ω} and intuitionistic derivability instead of PA^{ω} and classical derivability. However, there seems to be no other solution to prove that the predicate $V(\alpha, \delta)$ from the proof above is open than to use classical logic.[3]

Moreover, we have added extensionality in the verifying system in order to make the identification concerning list of pairs. If one adds η-equality, it should be possible to remove extensionality.

Definition 3.8. A predicate $U(\alpha)$ of the form

$$C_{\forall}(\alpha) \to B^{\exists}(\alpha)$$

is called *pointwise open*, where C is a predicate of arity $(\mathbf{N}, \rho)$, B a Σ-predicate of arity (ρ^*), and $C_{\forall}(\alpha) := \forall n C(n, \alpha n)$. *Weak update induction* $\widetilde{\mathrm{UI}}$ (*weak extended update induction* $\widetilde{\mathrm{EUI}}$) is UI (respectively EUI) restricted to pointwise open predicates.

Analogously to Lemma 3.1 one proves:

Lemma 3.9. $\mathsf{HA}^{\omega} + \widetilde{\mathrm{UI}}$ *and* $\mathsf{HA}^{\omega} + \widetilde{\mathrm{EUI}}$ *are closed under* Σ*-translation.*

Theorem 3.10. *For* $\Box \in \{\mathrm{UI}, \mathrm{EUI}\}$ *and any* Σ*-formula* B *we have:*

(1) $\mathsf{HA}^{\omega} + \widetilde{\Box} \vdash_{\mathrm{m}} \mathrm{AC}^{\mathrm{g}}$*, and*
(2) $\mathsf{PA}^{\omega} + \mathrm{AC} \vdash_{\mathrm{c}} \forall\vec{x}B$ *implies* $\mathsf{HA}^{\omega} + \widetilde{\Box} \vdash_{\mathrm{i}} \forall\vec{x}B$.

Proof. We only have to prove the statement for $\widetilde{\mathrm{UI}}$, as $\widetilde{\mathrm{EUI}}$ implies $\widetilde{\mathrm{UI}}$ in HA^{ω} (this is proved along the lines of the proof of Lemma 3.5 in Chapter 3). For (1) assume

[3] I am grateful to Ulrich Berger for this proposal.

(i) $\forall n \neg \forall x^\rho \neg A(n,x)$, and
(ii) $\forall f^{\mathbf{N}\to\rho} \neg \forall n A(n, fn)$.

We have to prove $\bot$. For $\alpha \colon (\mathbf{N}\times\rho)^\omega$ define

$$C(\alpha) := \forall n \in \mathsf{dom}(\alpha) A(n, \alpha[n]) \quad \text{and} \quad U(\alpha) := C(\alpha) \to \bot.$$

Observe that U is pointwise open and we have $\neg\forall\alpha U(\alpha)$ (e.g., $\neg U(0^{(\mathbf{N}\times\rho)^\omega})$). Thus by $\widetilde{\mathrm{UI}}$, it suffices to prove

$$\forall\alpha\big(\forall n \forall x (n \notin \mathsf{dom}(\alpha) \to U(\alpha_n^x)) \to U(\alpha)\big).$$

Fix α with

$$\forall n \forall x (n \notin \mathsf{dom}(\alpha) \to U(\alpha_n^x)), \text{ and} \tag{49}$$

$$C(\alpha). \tag{50}$$

To show: $\bot$. By (50) and (ii), it suffices to show that α is total. Let $n \colon \mathbf{N}$ and suppose $n \notin \mathsf{dom}(\alpha)$. By (i), it is enough to prove $\neg A(n,x)$ for all x^ρ. But $A(n,x)$ and $C(\alpha)$ yield $C(\alpha_n^x)$. Therefore $\bot$ by (49). This finishes the proof of (1).

For (2) suppose $\mathsf{PA}^\omega + \mathrm{AC} \vdash_{\mathrm{c}} B$. By negative translation, we get $\mathsf{HA}^\omega + \mathrm{AC}^{\mathrm{g}} \vdash_{\mathrm{m}} B^{\mathrm{g}}$, and hence $\mathsf{HA}^\omega + \widetilde{\mathrm{UI}} \vdash_{\mathrm{i}} B^{\mathrm{g}}$ by (1). But B is $\to\forall$-free, so Lemma 1.5 yields $\mathsf{HA}^\omega + \widetilde{\mathrm{UI}} \vdash_{\mathrm{i}} \neg\neg B$. Now the claim follows from Lemma 1.6 (1) using that $\mathsf{HA}^\omega + \widetilde{\mathrm{UI}}$ is closed under Σ-translation by Lemma 3.9. □

REMARK 3.11. Similarly to our pointwise open predicates, Berger [**4**] considers 1- and 2-open predicates. These have the property that update induction restricted to 1-open predicates (2-open predicates) proves the negative translation of AC (DC) in HA^ω.

THEOREM 3.12. *There exists a closed term t such that*

$$\mathsf{E\text{-}PA}^\omega + \mathrm{EUI} + \mathrm{Cont} + \mathrm{EUR}(\mathcal{R}^{\mathsf{e}}) \vdash_{\mathrm{c}} t(\mathcal{R}^{\mathsf{e}})\, \mathbf{mr}\, \widetilde{\mathrm{EUI}}.$$

PROOF. The proof is similar to the proof of Theorem 3.6. Let

$$U(\alpha^{(\mathbf{N}\times\rho)^\omega}) = C_\forall(\alpha) \to B^\exists(\alpha)$$

be pointwise open with a Σ-formula B. Set $\tau := \tau(B)$ and $\sigma := \tau(C)$.

Let's identify $(\mathbf{N}\times(\rho\times\sigma))^\omega$ with $(\mathbf{N}\times\rho)^\omega \times \sigma^\omega$, i.e., partial sequences of type $\rho\times\sigma$ with pairs of the form $\langle\alpha,\delta\rangle \colon (\mathbf{N}\times\rho)^\omega\times\sigma^\omega$ (so α is a partial sequence of type ρ). We also need the following notation for $\delta,\eta \colon \sigma^\omega$ and $\alpha \colon (\mathbf{N}\times\rho)^\omega$:

$$\delta\{\eta\}_\alpha(n) := \begin{cases} \delta n & \text{if } n \in \mathsf{dom}(\alpha), \\ \eta n & \text{otherwise.} \end{cases}$$

So with the identification from above, we obtain

$$\langle\alpha,\delta\rangle\{\langle\beta,\eta\rangle\} = \langle\alpha\{\beta\}, \delta\{\eta\}_\alpha\rangle, \text{ and } \langle\alpha,\delta\rangle(n) \ll \langle\beta,\eta\rangle(n) \text{ iff } \alpha n \ll \beta n.$$

Using extended update recursion we can define a functional Ψ satisfying

$$\Psi F \alpha\delta =_{\mathbf{N}\times\tau} F\alpha\big(\lambda n,\beta,\eta.\mathbf{if}\ \alpha n \ll \beta n\ \mathbf{then}\ \Psi F(\alpha\{\beta\})(\delta\{\eta\}_\alpha)\big)\delta.$$

We now prove that Ψ realizes weak extended update induction. Let F be a realizer for the premise, i.e.,

$$F\, \mathbf{mr}\, \forall\alpha\big[\forall n \forall\beta\big(\alpha n \ll \beta n \to U(\alpha\{\beta\})\big) \to U(\alpha)\big]. \tag{51}$$

Define $V(\alpha,\delta) := \delta\,\mathbf{mr}\,C_\forall(\alpha) \to \Psi F\alpha\delta\,\mathbf{mr}\,B^\exists(\alpha)$. As in Theorem 3.6, we conclude that V is open[4] so we can argue by EUI. Fix α and δ with

$$\delta\,\mathbf{mr}\,C_\forall(\alpha),\ \text{i.e., } \forall n\,\delta n\,\mathbf{mr}\,C(n,\alpha n). \tag{52}$$

To show: $\Psi F\alpha\delta\,\mathbf{mr}\,B^\exists(\alpha)$. Equation (51) unfolds to

$$\forall \alpha,\xi\left[\forall n,\beta\big(\alpha n \ll \beta n \to \xi n\beta\,\mathbf{mr}\,U(\alpha\{\beta\})\big) \to F\alpha\xi\,\mathbf{mr}\,U(\alpha)\right]. \tag{53}$$

Together with (52) and the definition of Ψ, this amounts to proving for all n and β with $\alpha n \ll \beta n$:

$$(\lambda\eta.\Psi f(\alpha\{\beta\})(\delta\{\eta\}_\alpha))\,\mathbf{mr}\,U(\alpha\{\beta\}).$$

So suppose $\alpha n \ll \beta n$ and

$$\eta\,\mathbf{mr}\,C_\forall(\alpha\{\beta\}),\ \text{i.e., } \forall k\,\eta(k)\,\mathbf{mr}\,C(k,\alpha\{\beta\}(k)). \tag{54}$$

By the extended update IH we have

$$\forall \varphi,\psi,m\big(\alpha m \ll \varphi m \to V(\alpha\{\varphi\},\delta\{\psi\}_\alpha)\big),$$

in particular, $V(\alpha\{\beta\},\delta\{\eta\}_\alpha)$. So it is enough to prove $\delta\{\eta\}_\alpha\,\mathbf{mr}\,C_\forall(\alpha\{\beta\})$, i.e., $\delta\{\eta\}_\alpha(k)\,\mathbf{mr}\,C(k,\alpha\{\beta\}(k))$ for all k. For $k \in \mathsf{dom}(\alpha)$ this follows from (52). Otherwise, if $k \notin \mathsf{dom}(\alpha)$, it follows from (54). □

Analogously to the last theorem one can also prove:

THEOREM 3.13. *There exists a closed term t such that*

$$\mathsf{E\text{-}PA}^\omega + \mathrm{UI} + \mathrm{Cont} + \mathrm{UR}(\mathcal{R}^{\mathsf{u}}) \vdash_{\mathrm{c}} t(\mathcal{R}^{\mathsf{u}})\,\mathbf{mr}\,\widetilde{\mathrm{UI}}.$$

THEOREM 3.14. *Let $A(x^\rho, y^\tau)$ be a Σ-formula with* $\mathrm{lev}(\tau) = 0$.

(i) *If $\mathsf{PA}^\omega + \mathrm{OI} \vdash_{\mathrm{c}} \forall x\exists y A(x,y)$, then there exists a term $\Phi^{\rho\to\tau}$ with free variables among $\vec{\Psi}$ such that*

$$\mathsf{E\text{-}PA}^\omega + \mathrm{Cont} + \mathrm{OI} + \mathrm{OR}(\vec{\Psi}) \vdash_{\mathrm{c}} \forall x A(x,\Phi x).$$

(ii) *If $\mathsf{PA}^\omega + \mathrm{AC} \vdash_{\mathrm{c}} \forall x\exists y A(x,y)$, then there exist terms $\Phi_1^{\rho\to\tau}$ and $\Phi_2^{\rho\to\tau}$ with $\mathsf{FV}(\Phi_i) \subseteq \vec{\Psi}_i$ such that:*

$$\mathsf{E\text{-}PA}^\omega + \mathrm{Cont} + \mathrm{EUI} + \mathrm{EUR}(\vec{\Psi}_1) \vdash_{\mathrm{c}} \forall x A(x,\Phi_1 x),\ \textit{and}$$

$$\mathsf{E\text{-}PA}^\omega + \mathrm{Cont} + \mathrm{UI} + \mathrm{UR}(\vec{\Psi}_2) \vdash_{\mathrm{c}} \forall x A(x,\Phi_2 x).$$

PROOF. (i). Suppose $\mathsf{PA}^\omega + \mathrm{OI} \vdash_{\mathrm{c}} \forall x\exists y A(x,y)$. Because of $\exists y A \in \Sigma$, by Corollary 3.2 we obtain $\mathsf{HA}^\omega + \mathrm{OI} \vdash_{\mathrm{i}} \forall x\exists y A(x,y)$. The Soundness Theorem for realizability yields a term t with free variables among $\vec{x}$ such that $\mathsf{HA}^\omega + \vec{x}\,\mathbf{mr}\,\mathrm{OI} \vdash_{\mathrm{i}} t\,\mathbf{mr}\,\forall x\exists y A(x,y)$,[5] i.e.,

$$\mathsf{HA}^\omega + \vec{x}\,\mathbf{mr}\,\mathrm{OI} \vdash_{\mathrm{i}} \forall x\,\pi_1(tx)\,\mathbf{mr}\,A(x,\pi_0(tx)).$$

By Lemma 3.4 (ii), we get

$$\mathsf{HA}^\omega + \vec{x}\,\mathbf{mr}\,\mathrm{OI} \vdash_{\mathrm{i}} \forall x\,A(x,\pi_0(tx)),$$

and thus applying Theorem 3.6 gives us

$$\mathsf{E\text{-}PA}^\omega + \mathrm{Cont} + \mathrm{OI} + \mathrm{OR}(\vec{\Psi}) \vdash_{\mathrm{c}} \forall x A(x,\Phi x)$$

[4]Note that V is not necessarily pointwise open.

[5]Here $\vec{x}\,\mathbf{mr}\,\mathrm{OI}$ means $\vec{x}\,\mathbf{mr}\,\vec{\mathrm{OI}}$ where $\vec{\mathrm{OI}}$ are all instances of OI used in the proof.

for a term Φ resulting from $\lambda x.\pi_0(tx)$ by replacing $\vec{x}$ with the appropriate realizers of open induction (which have all free variables among $\vec{\Psi}$).

(ii). Suppose $\mathsf{PA}^\omega + \mathrm{AC} \vdash_{\mathrm{c}} \forall x \exists y A(x,y)$. Then by Theorem 3.10 (2) $\mathsf{HA}^\omega + \widetilde{\mathrm{EUI}}(\widetilde{\mathrm{UI}}) \vdash_{\mathrm{i}} \forall x \exists y A(x,y)$. The rest is proved analogously to (i) using Theorem 3.12 (3.13 respectively). □

4. Realizing AC$^{\mathrm{g}}$ and DC^{g} with Modified Bar Recursion

In this section we show, following [**6**], how to obtain a realizer of the negative translations of dependent choice and countable choice via modified bar recursion. Instead of directly realizing the negative translated choice axioms, we follow Spector [**42**], and reduce them to the so called *double negation shift*

$$\mathrm{DNS} \qquad \forall n \neg\neg A \to \neg\neg \forall n A$$

together with the non-translated axioms. One easily sees that

$$\mathrm{AC} + \mathrm{DNS} \vdash_{\mathrm{m}} \mathrm{AC}^{\mathrm{g}} \quad \text{and} \quad \mathrm{DC} + \mathrm{DNS} \vdash_{\mathrm{m}} \mathrm{DC}^{\mathrm{g}}.$$

To verify the correctness of our realizer, we need the following principle.

Definition 4.1. *Relativized quantifier-free bar induction* $\mathrm{rBI}_{\mathsf{qf}}$ is the following scheme

$$\begin{aligned}
&\forall \alpha \in S \exists n P(\bar{\alpha} n) \wedge \\
&\forall s \in S\big(\forall x^\rho [S(s*x) \to P(s*x)] \to P(s)\big) \wedge \\
&S(\langle\rangle) \ \to \\
&P(\langle\rangle),
\end{aligned}$$

where $S(s)$ is an arbitrary and $P(s)$ a quantifier-free predicate. We have written $\alpha \in S$ for $\forall n S(\bar{\alpha} n)$, and $s \in S$ for $S(s)$. In fact, we will only use a special case called *relativized quantifier-free pointwise bar induction* $\mathrm{rBI}^{\mathsf{pt}}_{\mathsf{qf}}$, which is given by

$$\begin{aligned}
&\forall \alpha \in S \exists n P(\bar{\alpha} n) \wedge \\
&\forall s \in S\big(\forall x^\rho [S(x,|s|) \to P(s*x)] \to P(s)\big) \ \to \\
&P(\langle\rangle)
\end{aligned}$$

where $S(x^\rho, n)$ is an arbitrary and $P(s)$ a quantifier-free predicate. Here $\alpha \in S$ stands for $\forall n S(\alpha n, n)$, and $s \in S$ for $\forall i < |s| S(s_i, i)$.

Lemma 4.2. *There exists a closed term H such that*

$$\mathsf{HA}^\omega \vdash_{\mathrm{m}} \forall \vec{x}\, H \,\mathbf{mr}\, (A \to B_A),$$

where the free variables of $A \to B_A$ are among $\vec{x}$.

Proof. By Lemma 1.4 (1) $\vdash_{\mathrm{m}} A \to B_A$. Moreover, this can be derived without the use of $\forall$-elimination rules and with $\exists$-introduction on closed terms only[6]. Hence by Theorem 2.3, we obtain a closed term H such that $\mathsf{HA}^\omega \vdash_{\mathrm{m}} H \,\mathbf{mr}\, (A \to B_A)$, and thus the claim follows. □

[6]This is proved by induction on B, where in the case $B = \exists x^\rho C(x)$ ($x \notin \mathsf{FV}(A)$) the IH yields $\vdash_{\mathrm{m}} A \to C_A(x)$. Hence substituting 0^ρ for x in the proof gives $\vdash_{\mathrm{m}} A \to C_A(0)$, and thus $\vdash_{\mathrm{m}} A \to B_A$, where the $\exists$-introduction is with the term 0^ρ.

It is crucial that H in the last lemma is closed, i.e., does not depend on the variables $\vec{x}$.

THEOREM 4.3. *Let A be a formula with $\tau(A) = \mathbf{N}$. Then for each instance of the A-translated double negation shift* DNS_A*, there exists a term t with the only free variable Φ, such that*

$$\mathsf{HA}^\omega + \mathrm{rBI}^{\mathsf{pt}}_{\mathsf{qf}} + \mathrm{wMBR}(\Phi) + \mathrm{Cont} \vdash_{\mathrm{m}} t\,\mathbf{mr}\,\mathrm{DNS}_A.$$

PROOF. Consider an instance of DNS

$$\forall n \neg\neg C(n) \to \neg\neg\forall n C(n).^7$$

Then its A-translation is given by

$$\forall n((C_A(n) \to A) \to A) \to (\forall n C_A(n) \to A) \to A.$$

Let $B := C_A$ and $\rho := \tau(B)$. Assume we have realizers G and Y for the premises, i.e.,

$$G\,\mathbf{mr}\,\forall n\big((B(n) \to A) \to A\big) \quad \text{and} \quad Y\,\mathbf{mr}\,\big(\forall n B(n) \to A\big). \tag{55}$$

with the types

$$G\colon \mathbf{N} \to (\rho \to \mathbf{N}) \to \mathbf{N} \quad \text{and} \quad Y\colon \rho^\omega \to \mathbf{N}.$$

We have to find a realizer for A. By Lemma 4.2, there is a closed term H with

$$\forall n\, H\,\mathbf{mr}\,(A \to B(n)). \tag{56}$$

Using wMBR we can define Ψ as:

$$\Psi Y G s =_{\mathbf{N}} Y\big(s \mathbin{@} \lambda k.H(G(|s|, \lambda x.\Psi Y G(s * x)))\big). \tag{57}$$

For readability, let's omit the arguments Y and G from Ψ. We now prove that $\Psi\langle\rangle\,\mathbf{mr}\,A$ using $\mathrm{rBI}^{\mathsf{pt}}_{\mathsf{qf}}$ on the predicates

$$S(x, n) :\leftrightarrow x\,\mathbf{mr}\,B(n),$$
$$P(s) :\leftrightarrow \Psi s\,\mathbf{mr}\,A.$$

We verify the requirements for $\mathrm{rBI}^{\mathsf{pt}}_{\mathsf{qf}}$:

(i) $\forall \alpha \in S \exists n P(\bar{\alpha} n)$. Let $\alpha \in S$ and n be a point of continuity of Y at α. We have

$$\alpha \in S \leftrightarrow \forall n\, \alpha n\,\mathbf{mr}\,B(n) \leftrightarrow \alpha\,\mathbf{mr}\,\forall n B(n).$$

Hence by (55) $Y(\alpha)\,\mathbf{mr}\,A$. But $\Psi(\bar{\alpha} n) = Y(\bar{\alpha} n \mathbin{@} \dots) = Y(\alpha)$ by continuity.

[7] Of course, the actual instance of DNS is the universal closure of the displayed formula. To obtain a realizer for the universal closure one simply lambda abstracts the corresponding variables in the realizer below. The realizer does not contain these variables free, and hence the universal quantifiers of the universal closure may be regarded as "computationally irrelevant".

(ii) $\forall s \in S\big(\forall x^\rho[S(x,|s|) \to P(s*x)] \to P(s)\big)$. Let $s \in S$ with

$$\forall x\big(S(x,|s|) \to P(s*x)\big).$$

We have to show $P(s)$. We obtain:

$$\begin{aligned}
&\forall x\big(x\,\mathbf{mr}\,B(|s|) \to \Psi(s*x)\,\mathbf{mr}\,A\big) \\
&\to \lambda x.\Psi(s*x)\,\mathbf{mr}\,\big(B(|s|) \to A\big) \\
&\to G(|s|,\lambda x.\Psi(s*x))\,\mathbf{mr}\,A && \text{by (55)} \\
&\to \lambda k.H(G(|s|,\lambda x.\Psi(s*x)))\,\mathbf{mr}\,\forall n B(n) && \text{by (56)} \\
&\to s \mathbin{@} \lambda k.H(G(|s|,\lambda x.\Psi(s*x)))\,\mathbf{mr}\,\forall n B(n) && \text{since } s \in S \\
&\to Y\big(s \mathbin{@} \lambda k.H(G(|s|,\lambda x.\Psi(s*x)))\big)\,\mathbf{mr}\,A && \text{by (55)} \\
&\to \Psi(s)\,\mathbf{mr}\,A && \text{by (57)} \\
&\to P(s).
\end{aligned}$$

By $\mathrm{rBI}^{\mathsf{pt}}_{\mathsf{qf}}$ we conclude $P(\langle\rangle)$, i.e., $\Psi\langle\rangle\,\mathbf{mr}\,A$. □

LEMMA 4.4. *There are closed terms t and t' such that:*
(i) $\mathsf{HA}^\omega \vdash_{\mathrm{m}} t\,\mathbf{mr}\,\mathrm{AC}$,
(ii) $\mathsf{HA}^\omega \vdash_{\mathrm{m}} t'\,\mathbf{mr}\,\mathrm{DC}$.

PROOF. (i). By the definition of modified realizability:

$$\begin{aligned}
&t\,\mathbf{mr}\,\big(\forall n\exists x A(n,x) \to \exists f\forall n A(n,f(n))\big) \\
&= \forall z\big(\forall n\, \pi_1(zn)\,\mathbf{mr}\,A(n,\pi_0(zn)) \to \forall n\, \pi_1(tz)n\,\mathbf{mr}\,A(n,\pi_0(tz)n)\big).
\end{aligned}$$

Hence we can take $\lambda z.\langle \lambda n.\pi_0(zn), \lambda n.\pi_1(zn)\rangle$ for t.

(ii). Similar for DC:

$$\begin{aligned}
&t\,\mathbf{mr}\,\big(\forall n\forall x^\rho\exists^\rho y A(n,x,y) \to \exists f\forall n A(n,f(n),f(n+1))\big) \\
&= \forall z\big(\forall n\forall x\, \pi_1(znx)\,\mathbf{mr}\,A(n,x,\pi_0(znx)) \to \\
&\qquad \forall n\, \pi_1(tz)\,\mathbf{mr}\,A(n,\pi_0(tz)(n),\pi_0(tz)(n+1))\big).
\end{aligned}$$

Define φ by

$$\varphi z0 := 0^\rho \text{ and } \varphi z(n+1) := \pi_0(zn(\varphi zn)).$$

We leave it to the reader to verify that $t := \lambda z.\langle \varphi z, \lambda n.\pi_1(zn(\varphi zn))\rangle$ realizes dependent choice. □

THEOREM 4.5. *Let $A(x^\rho, n^{\mathbf{N}})$ be an atomic formula and suppose*

$$\mathsf{PA}^\omega + \mathrm{DC} \vdash_{\mathrm{c}} \forall x^\rho \exists n A(x,n).$$

Then there exists a term t with free variables among $\vec{\Phi}$ such that

$$\mathsf{HA}^\omega + \mathrm{rBI}^{\mathsf{pt}}_{\mathsf{qf}} + \mathrm{wMBR}(\vec{\Phi}) + \mathrm{Cont} \vdash_{\mathrm{m}} \forall x^\rho A(x,tx).$$

PROOF. Suppose we have a derivation

$$\mathsf{PA}^\omega + \mathrm{DC} \vdash_{\mathrm{c}} \forall x^\rho \exists n A(x,n).$$

Applying the negative translation yields

$$\mathsf{HA}^\omega + \mathrm{DC}^{\mathrm{g}} \vdash_{\mathrm{m}} \forall x \neg\forall n \neg\neg\neg A(x,n)$$

and hence

$$\mathsf{HA}^\omega + \mathrm{DNS} + \mathrm{DC} \vdash_{\mathrm{m}} \neg\forall n \neg A(x,n).$$

Set $B(x) := \exists n A(x,n)$. Now B-translation yields

$$\mathsf{HA}^\omega + \mathrm{DNS}_B + \mathrm{DC} \vdash_{\mathrm{m}} \forall n\big(A(x,n) \lor B(x) \to B(x)\big) \to B(x).$$

Clearly, $A(x,n) \lor B(x) \to B(x)$ is derivable and thus

$$\mathsf{HA}^\omega + \mathrm{DNS}_B + \mathrm{DC} \vdash_{\mathrm{m}} \exists n A(x,n).$$

Let $\forall\mathrm{DNS}_B$ be the universal closures of the formulas in DNS_B. We conclude

$$\mathsf{HA}^\omega + \forall\mathrm{DNS}_B + \mathrm{DC} \vdash_{\mathrm{m}} \forall x \exists n A(x,m).$$

Note that $\tau(B) = \mathbf{N}$. Now, the Soundness Theorem for realizability, together with Theorem 4.3 and Lemma 4.4 yield a term t with free variables among $\vec{\Phi}$ such that

$$\mathsf{HA}^\omega + \mathrm{rBI}^{\mathsf{pt}}_{\mathsf{qf}} + \mathrm{wMBR}(\vec{\Phi}) + \mathrm{Cont} \vdash_{\mathrm{m}} t\,\mathbf{mr}\,\forall x \exists n A(x,n).$$

By definition, $t\,\mathbf{mr}\,\forall x \exists n A(x,n)$ is equal to $\forall x A(x,tx)$. This finishes the proof. $\square$

5. Notes

It should be noted that one can in fact compute witnesses with our realizers. If one enriches Gödel's T with defined constants and computation rules according to the scheme one wants to use (e.g., a variant of open recursion or modified bar recursion), the Adequacy Theorem of Chapter 2 ensures that we can effectively compute (with our operational semantics) any term of type $\mathbf{N}$. (Cf. Remark 1.4 in Chapter 3.)

From the point of view of reductive proof theory, the results from the last two sections are rather limited: In Theorem 3.14, the verifying systems of our realizers are classical and comprise the axiom of choice. In Theorem 4.5, the verifying system contains $\mathsf{HA}^\omega + \mathrm{rBI}^{\mathsf{pt}}_{\mathsf{qf}}$ — a theory where AC^{g} is easily derivable and thus, using negative translation, $\mathsf{PA}^\omega + \mathrm{AC} \vdash A$ implies $\mathsf{HA}^\omega + \mathrm{rBI}^{\mathsf{pt}}_{\mathsf{qf}} \vdash A^{\mathrm{g}}$ for each formula A. Compared to this, Spector's reduction to a *quantifier-free* system seems to be a more "foundational" result (although questionable from a constructive point of view, cf. [**1**, p. 370 f.]).

Nevertheless, from the point of view of program extraction, the emphasis lies on correct and usable algorithms. For this, the methods from the last to sections seem to be more direct than the interpretation of Spector which uses Dialectica interpretation instead of the more direct realizability interpretation. A case study of a non-trivial example using modified bar recursion together with an implementation in the proof assistant Minlog[8] has been done by Seisenberger in [**41**].

[8]See `http://www.minlog-system.de`

Bibliography

1. Jeremy Avigad and Solomon Feferman, *Gödel's functional ("Dialectica") interpretation*, Handbook of Proof Theory (S. Buss, ed.), Stud. Logic Found. Math., vol. 137, North–Holland, Amsterdam, 1999, pp. 337–405.
2. Stefano Berardi, Marc Bezem, and Thierry Coquand, *On the computational content of the axiom of choice*, The Journal of Symbolic Logic **63** (1998), no. 2, 600–622.
3. Ulrich Berger, *Totale Objekte und Mengen in der Bereichstheorie*, Ph.D. thesis, Mathematisches Institut der Universität München, 1990.
4. ———, *A computational interpretation of open induction*, Proceedings of the Nineteenth Annual IEEE Symposium on Logic in Computer Science (F. Titsworth, ed.), IEEE Computer Society, 2004, pp. 326–334.
5. ———, *Continuous semantics for strong normalization*, New Computational Paradigms, First Conference on Computability in Europe, CiE 2005, Amsterdam, The Netherlands, June 8-12, 2005, Proceedings (S.B. Cooper, B. Löwe, and L. Torenvliet, eds.), Lecture Notes in Computer Science, no. 3526, Springer Verlag, Berlin, Heidelberg, New York, 2005, pp. 23–34.
6. Ulrich Berger and Paulo Oliva, *Modified bar recursion and classical dependent choice*, Logic Colloquium '01, Proceedings of the Annual European Summer Meeting of the Association for Symbolic Logic, held in Vienna, Austria, August 6 - 11, 2001 (M. Baaz, S.D. Friedman, and J. Kraijcek, eds.), Lecture Notes in Logic, vol. 20, Springer Verlag, Berlin, Heidelberg, New York, 2005, pp. 89–107.
7. ———, *Modified bar recursion*, Mathematical Structures in Computer Science **16** (2006), 163–183.
8. Marc Bezem, *Strongly majorizable functionals of finite type: a model for barrecursion containing discontinuous functionals*, The Journal of Symbolic Logic **50** (1985), 652–660.
9. ———, *Equivalence of bar recursors in the theory of functionals of finite type*, Archive for Mathematical Logic **27** (1988), 149–160.
10. Thierry Coquand, *Constructive topology and combinatorics*, Constructivity in Computer Science (J. Paul Myers Jr. and Michael J. O'Donnell, eds.), Lecture Notes in Computer Science, vol. 613, Springer Verlag, Berlin, Heidelberg, New York, 1992, pp. 159–164.
11. Albert Dragalin, *New kinds of realizability*, Abstracts of the 6th International Congress of Logic, Methodology and Philosophy of Sciences (Hannover, Germany), 1979, pp. 20–24.
12. Yuri L. Ershov, *Model C of the partial continuous functionals*, Logic Colloquium 1976 (R. Gandy and M. Hyland, eds.), North–Holland, Amsterdam, 1977, pp. 455–467.
13. Martín H. Escardó, *Exhaustible sets in higher-type computation*, Logical Methods in Computer Science **4** (2008), no. 3:3, 1–37.
14. Martín H. Escardó and Paulo Oliva, *On variants of modified bar recursion*, Slides of a talk at Domains IX, Brighton, September 2008.
15. ———, *Selection functions, bar recursion, and backward induction*, To appear in Mathematical Structures in Computer Science, 2009.
16. Harvey Friedman, *Classically and intuitionistically provably recursive functions*, Higher Set Theory (D.S. Scott and G.H. Müller, eds.), Lecture Notes in Mathematics, vol. 669, Springer Verlag, Berlin, Heidelberg, New York, 1978, pp. 21–28.
17. Jean-Yves Girard, *Interprétation functionelle et élimination des coupures de l'arithmétique d'ordre supérieur*, Thèse de Doctorat d'État, Université Paris VII, 1972.

18. Kurt Gödel, *Über eine bisher noch nicht benützte Erweiterung des finiten Standpunkts*, Dialectica **12** (1958), 280–287.
19. ______, *On a hitherto unexploited extension of the finitary standpoint*, Journal of Philosophical Logic **9** (1980), no. 2, 133–142.
20. David Hilbert, *Über das Unendliche*, Mathematische Annalen **95** (1926), 161–190.
21. William A. Howard, *Functional interpretation of bar induction by bar recursion*, Compositio Mathematica **20** (1968), 107–124.
22. William A. Howard and Georg Kreisel, *Transfinite induction and bar induction of types zero and one, and the role of continuity in intuitionistic analysis*, The Journal of Symbolic Logic **31** (1966), no. 3, 325–358.
23. Stephen C. Kleene, *Countable functionals*, Constructivity in Mathematics (A. Heyting, ed.), North–Holland, Amsterdam, 1959, pp. 81–100.
24. Ulrich Kohlenbach, *Theorie der majorisierbaren und stetigen Funktionale und ihre Anwendung bei der Extraktion von Schranken aus inkonstruktiven Beweisen: effektive Eindeutigkeitsmodule bei besten Approximationen aus ineffektiven Eindeutigkeitsbeweisen*, Ph.D. thesis, Johann Wolfgang Goethe-Universität, Fachber. Math., Frankfurt am Main, 1990.
25. Georg Kreisel, *Mathematical significance of consistency proofs*, The Journal of Symbolic Logic **23** (1958), no. 2, 155–182.
26. ______, *Interpretation of analysis by means of constructive functionals of finite types*, Constructivity in Mathematics (A. Heyting, ed.), North–Holland, Amsterdam, 1959, pp. 101–128.
27. Guiseppe Longo and Eugenio Moggi, *The hereditary partial effective functionals and recursion theory in higher types*, The Journal of Symbolic Logic **49** (1984), no. 4, 1319–1331.
28. Per Martin-Löf, *The domain interpretation of type theory*, Talk at the workshop on semantics of programming languages, Chalmers University, Göteborg, August 1983.
29. Robin Milner, *Models of LCF*, Technical Report Memo Aim-186, Stanford Artificial Intelligence Laboratory, January 1973.
30. C.St.J.A. Nash-Williams, *On well-quasi-ordering finite trees*, Proc. Cambridge Philosophical Society **59** (1963), 833–835.
31. Dag Normann, *The continuous functionals*, Handbook of Computability Theory (E. Griffor, ed.), Stud. Logic Found. Math., vol. 140, North–Holland, Amsterdam, 1999, pp. 251–275.
32. Erik Palmgren, *An intuitionistic axiomatisation of real closed fields*, Mathematical Logic Quarterly **48** (2002), no. 2, 297–299.
33. Gordon D. Plotkin, *LCF considered as a programming language*, Theoretical Computer Science **5** (1977), 223–255.
34. Jean-Claude Raoult, *Proving open properties by induction*, Information Processing Letters **29** (1988), 19–23.
35. Helmut Schwichtenberg, *Proof theory*, Notes from a lecture course, Universität München, 2006.
36. ______, *Recursion on the partial continuous functionals*, Logic Colloquium 2005 (C. Dimitracopoulos, L. Newelski, D. Normann, and J. Steel, eds.), Lecture Notes in Logic, vol. 28, Association for Symbolic Logic, 2007, pp. 173–201.
37. Helmut Schwichtenberg and Stanley S. Wainer, *Proofs and computations*, Book-manuscript, to appear: Springer Verlag, 2009.
38. Dana S. Scott, *A type-theoretical alternative to ISWIM, CHUCH, OWHY*, Manuscript, Oxford University, published as [**40**], 1969.
39. ______, *Domains for denotational semantics*, Proceedings of the 9th Colloquium on Automata, Languages and Programming (E. Nielsen and E.M. Schmidt, eds.), LNCS, vol. 140, Springer Verlag, Berlin, Heidelberg, New York, 1982, pp. 577–613.
40. ______, *A type-theoretical alternative to ISWIM, CHUCH, OWHY*, Theoretical Computer Science **121** (1993), no. 1-2, 441–440.
41. Monika Seisenberger, *Programs from proofs using classical dependent choice*, Annals of Pure and Applied Logic **158** (2008), no. 1-3, 90–110.

42. Clifford Spector, *Provably recursive functionals of analysis: a consistency proof of analysis by an extension in current intuitionistic mathematics*, Recursive Function Theory (F.D.E. Dekker, ed.), Proc. Symposia in Pure Mathematics, vol. 5, American Mathematical Society, Providence, Rhode Island, 1962, pp. 1–27.
43. Viggo Stoltenberg-Hansen, Ingrid Lindström, and Edward Griffor, *Mathematical theory of domains*, Cambridrige Tracts in Theoretical Computer Science, vol. 22, Cambridge University Press, 1994.
44. William W. Tait, *Normal form theorem for bar recursive functions of finite type*, Proceedings of the Second Scandinavian Logic Symposium (J.E. Fenstad, ed.), North–Holland, Amsterdam, 1971, pp. 353–367.
45. Masako Takahashi, *Parallel reductions in λ-calculus*, Information and Computation **118** (1995), 120–127.
46. Terese, *Term rewriting systems*, Cambridge University Press, 2003, Ed. by M. Bezem, J.W. Klop, and R. de Vrijer.
47. Anne S. Troelstra (ed.), *Metamathematical investigations of intuitionistic arithmetic and analysis*, Lecture Notes in Mathematics, vol. 344, Springer Verlag, Berlin, Heidelberg, New York, 1973.
48. Anne S. Troelstra and Helmut Schwichtenberg, *Basic proof theory*, 2nd ed., Cambridrige Tracts in Theoretical Computer Science, vol. 43, Cambridge University Press, 2000.
49. Helmut Vogel, *Ein starker Normalisationssatz für die bar-rekursiven Funktionale*, Archiv für Mathematische Logik und Grundlagenforschung **18** (1976), 81–84.

Index

Erklärung

Hiermit erkläre ich, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

München, den 14. Januar 2010

(Simon Huber)