A Hierarchy of Ramified Theories Below Primitive Recursive Arithmetic

Elliott John Spoors

Submitted in accordance with the requirements
for the degree of Doctor of Philosophy

The University of Leeds
Department of Pure Mathematics

September 2010

Acknowledgements

I wish to offer my sincere thanks to my supervisor Stan Wainer. He has been an inspirational and dedicated teacher who has guided the development of my passion for the subject of Mathematical Logic. I have taken a great deal of enjoyment from our many meetings over the years and thank him for all his assistance and encouragement. I also consider him a close friend and thank him for the patience and commitment he has shown to me during difficult times.

I also wish to thank the wider mathematical community from Leeds for providing a supportive, stimulating and friendly atmosphere to work within. Particular thanks to Barry Cooper and Michael Rathjen for their enthusiastic introductions to Computability Theory and Logic which motivated me to study further. Thanks are also due to the University of Leeds Scholarship funding which made this PhD a possibility.

To all my friends, in Leeds and elsewhere, a big thank you. In particular I wish to thank Eddy and Wan-Ley. You've been close friends of mine for many years now and have played a significant role in helping make my time at university so enjoyable. Thanks for sticking by me and putting up with me.

Finally I reserve a special thanks to my family. To my Mum and Dad for their continued support throughout my education. Their financial support has afforded me the opportunities to study but without their encouragement and emotional support this thesis would simply not have been possible. I owe you both a huge debt of gratitude. To my sister Jo, thank you for being such a good friend to me and for always having time to listen. I would not have got this far without you. Also thanks to my brother-in-law Jules and the newest members of the family, Sebastian and Alexander, for providing me with so much enjoyment and fun whenever I needed diversion from work.

Abstract

The arithmetical theory $EA(I;O)$ developed by Çağman, Ostrin and Wainer ([18] and [48]) provides a formal setting for the variable separation of Bellantoni-Cook predicative recursion [6]. As such, $EA(I;O)$ separates variables into outputs, which are quantified over, and inputs, for which induction applies. Inputs remain free throughout giving inductions in $EA(I;O)$ a pointwise character termed *predicative induction.* The result of this restriction is that the provably recursive functions are the elementary functions. An infinitary analysis brings out a connection to the Slow-Growing Hierarchy yielding ε_0 as the appropriate proof-theoretic ordinal in a pointwise sense. Chapters 1 and 2 are devoted to an exposition of these results.

In Chapter 3 a new principle of Σ_1-*closure* is introduced in constructing a conservative extension of $EA(I;O)$ named EA^1. This principle collapses the variable separation in $EA(I;O)$ and allows quantification over inputs by acting as an internalised ω-rule. EA^1 then provides a natural setting to address the problem of input substitution in ramified theories.

Chapters 4 and 5 introduce a hierarchy of theories based upon alternate additions of the predicative induction and Σ_1-closure principles. For $0 < k \in \mathbf{N}$, the provably recursive functions of the theories EA^k are shown to be the Grzegorczyk classes $\mathcal{E}^{k+2}$. Upper bounds are obtained via embeddings into appropriately layered infinitary systems with carefully controlled bounding functions for existential quantifiers. The theory $EA^{\prec\omega}$, defined by closure under finite applications of these two principles, is shown to be equivalent to primitive recursive arithmetic. The hierarchy generated may be considered as an implicit ramification of the sub-system of Peano Arithmetic which restricts induction to Σ_1-formulae.

Contents

Acknowledgements i

Abstract ii

Contents iii

Introduction **1**

1 *EA(I;O)* - An Elementary Arithmetic **8**

1.1 Preliminary Definitions 8

1.2 Basic Results 17

1.3 Exponentiation in $EA(I;O)$ 25

1.4 The Elementary Functions Are Provably Recursive in $EA(I;O)$ 33

2 An Infinitary Theory for *EA(I;O)* **37**

2.1 Introduction 37

2.2 Computations in $EA_\infty(I;O)$ 46

2.3 Structural Rules for $EA_\infty(I;O)$ 51

2.4 Cut-Elimination for $EA_\infty(I;O)$ 54

2.5 Embedding of $EA(I;O)$ 60

2.6 The Provably Recursive Functions of $EA(I;O)$ 65

3 *EA*^{1} - A Conservative Closure of *EA(I;O)* **70**

3.1 Introduction and Definitions for EA^1 70

3.2 Bounded Arithmetic and EA^1 76

3.3 An Infinitary Theory for EA^1 79

3.4 Computations in EA^1_∞ 81

3.5 Cut-Elimination for EA^1_∞ 84

3.6 Embedding of EA^1 89

3.7 EA^1 Is Σ_1 Conservative over $EA(I;O)$ 91

4 *EA*^{1}*(I;O)* and *EA*^{2} **95**

4.1 Introduction 95

4.2 Lower Bounds for Provably Recursive Functions 97

4.3 Upper Bounds for Provably Recursive Functions 104

    4.3.1 An Infinitary Theory for $EA^1(I;O)$ 104

    4.3.2 An Infinitary Theory for EA^2 115

5 The Hierarchy *EA*^{k} for *k* $\in$ N **120**

5.1 Basic Definitions 120

5.2 Lower Bounds for Provably Recursive Functions 121

5.3 Infinitary Theories for EA^k 122

5.4 The Theory $EA^{\prec\omega}$ 125

Appendix **126**

A. Derivations of Basic Results 126

B. Sub-Recursive Hierarchies 135

Bibliography **141**

Introduction

Grzegorczyk [29] introduced a proper hierarchy of classes of number-theoretic functions, $\mathcal{E}^k$ for $k \in \mathbf{N}$, whose union is the primitive recursive functions. The classes are stratified using bounded primitive recursion with bounds provided from a backbone hierarchy of strictly increasing functions E_k. The Grzegorczyk Hierarchy exhibits a strong stability under various definitions, cf. [43], such as rate of growth, computation limited by time or space bounds, enumeration, number of recursions and nesting of for-loops in loop programs. At the lower levels $\mathcal{E}^2$ consists of polynomially bounded functions and $\mathcal{E}^3$ the elementary functions which are bounded by fixed iterates of exponentiation. These classes may be seen as distinguishing between functions considered computationally feasible and infeasible in terms of their complexity.

Within the setting of formal theories of arithmetic one can seek to classify the strength of the theory in question by the assignment of an ordinal which seeks to measure complexity. This program of ordinal analysis dates back to Genzten [24] (translated in [25]) where transfinite induction up to the ordinal ε_0 sufficed to show the consistency of Peano Arithmetic. Considering the recursive functions definable in such theories the same ordinal re-appears. The origins of this approach stem from Kreisel [33] who showed the functions *computable* in Peano Arithmetic to be precisely those definable by recursions over well-orderings of $\mathbf{N}$ of order-types below ε_0. Later work by Löb and Wainer [37], [38],[39]; Parsons [50]; Schwichtenberg [55] and Wainer [61] for example, refined this result to sub-hierarchies of Peano Arithmetic. The Schwichtenberg-Wainer

Fast-Growing Hierarchy of functions provides a backbone of bounding functions in such analyses. (They extend the Grzegorczyk functions E_k by transfinite ordinal recursions.) A recent comprehensive treatment is given by Fairtlough and Wainer [19]. In all such cases the sub-hierarchies of Peano Arithmetic are given by restrictions on the complexity of formulae occurring in the induction axioms. A hierarchy of theories below primitive recursive arithmetic whose provably recursive functions correspond to the Grzegorczyk classes $\mathcal{E}^k$ may be given. One simply restricts inductions to bounded formulae and adds axioms expressing the totality of the Grzegorczyk function E_k, cf. [12]. We consider these explicit restrictions, using *a priori* bounds, as the *classical* approach which corresponds to defining function classes by bounded primitive recursion.

Simmons [57], and later independently Bellantoni and Cook [6], introduced a new approach to restricting primitive recursion by ramifying variables thus avoiding recourse to *a priori* bounds. The theme in so-called *predicative recursion* is to use two kinds of variables: normal and safe. One may use the usual (unbounded) primitive recursion scheme with the simple restriction that substitutions are only allowed on safe variables whilst normal variables admit the recursions. This predicative approach provides *implicit* bounds and naturally restricts the primitive recursive functions to those considered feasibly computable, for example $PTIME$ in [6]. The idea proved of great interest in theoretical computer science spawning a new paradigm of implicit computational complexity (also termed resource free or machine independent). Many complexity classes were defined using variable separation schemes, see for example Bellantoni [4]; Clote [13], [14]; Covino and Pani [16]; Leivant [35], [36] and Oitavem [44], [45]. The mantra ‘substitute at lower levels and recurse at a new higher level’ has been used by Bellantoni and Niggl [7], Caporaso et al. [9] and Wirz [67] to capture every Grzegorczyk class $\mathcal{E}^k$ by extending variable separation to an arbitrary finite number of levels. The general approach here exploits the idea of counting the depth of recursions used in generating primitive recursive functions, see Schwichtenberg [54].

The theory $EA(I;O)$ developed by Çağman, Ostrin and Wainer ([18], [46], [47], [48]) seeks to incorporate the Bellantoni-Cook/Simmons notion of predicative recursion into formal arithmetic. It builds upon previous ideas of Leivant's ramified theories [34]. $EA(I;O)$ possesses two levels of variables: the outputs over which quantifications (substitutions) apply and the inputs for which inductions (recursions) apply. The induction axiom used in $EA(I;O)$ may be termed 'predicative' since in the antecedent we are quantifying over numbers from an unrestricted domain of *values* (the outputs) whilst in the consequent we introduce a new variable x from a restricted domain of *numbers* (the inputs) which then remains free. As such these input variables may be more accurately termed *uninterpreted input constants* and induction in this context is pointwise. Importantly there are no restrictions on the complexity of the formulae allowed within inductions. Thus the implicit nature of Bellantoni-Cook recursion is retained in opposition to the classical approach to restricting complexity in arithmetic.

The analysis of $EA(I;O)$ shows that the functions provably recursive are Grzegorczyk's class $\mathcal{E}^3$, the Kalmár-Csillag ([17],[32]) elementary functions. Sub-hierarchies within $EA(I;O)$ may be generated by restricting the complexity of induction formulae reflecting the usual sub-hierarchies of Peano Arithmetic. This yields provably recursive functions from $\mathcal{E}^2$ and the exponential hierarchy between $\mathcal{E}^2$ and $\mathcal{E}^3$, [46]. In [48] it is noted that over a binary notation, natural characterizations of PTIME and EXP may be given similarly to the results of Leivant.

In analysing $EA(I;O)$ we may also see a reflection of the idea that, in both the function algebras and formal theories, variable separation is a process which collapses the classical fast-growing to slow-growing or feasible. Adopting the standard infinitary methods, for example in the analysis of Peano Arithmetic in [19], one finds bounding functions are now provided by the *Slow-Growing Hierarchy* and hence the associated proof-theoretic ordinal is ε_0 but in a pointwise sense, cf. [46] or [47]. (Collapsing the Slow-Growing Hierarchy below ε_0 yields the elementary functions.) Likewise, Wirz [68] shows pointwise transfinite induction up to ε_0 is provable in $EA(I;O)$. This lends support to the established idea that one may assign a theory a pointwise/slow-growing proof-theoretic ordinal, for example in the work of Arai [1], [2] and Schmerl [52]. Indeed these results, and the hierarchy comparison theorem of Girard [26], and later Wainer [62], are seen in the context of variable separation by the recent work of Wainer and Williams [63] and Williams [66]. There, $EA(I;O)$ is extended by finitely iterated inductive definitions and a new context is given in which the Slow-Growing Hierarchy 'catches up' with the Fast-Growing Hierarchy at the ordinal of $ID_{\prec\omega}$.

We offer some brief remarks on predicativity as it is a term frequently used in association with variable separation. The notion of predicativity dates back to Russel and Poincaré, see Feferman [20]. In proof theory the boundary between predicative and impredicative is usually considered to be those theories with proof-theoretic strength Γ_0. However, in such a context predicative is taken to mean 'predicative given the natural numbers'. Feferman [20] has claimed that predicativity should perhaps be seen as a *relative* (with respect to a given foundational scheme) rather than an *absolute* concept. Along such lines, a stricter finitistic notion of predicativity is given by Nelson [42]. Regarding the usual induction axioms of Peano Arithmetic as impredicative he argues:

> "It is not correct to argue that induction only involves the numbers from 0 to n; the property of n being established may be a formula with bound variables that are thought of as ranging over all numbers [....] A number is conceived to be an object satisfying every inductive formula; for a particular inductive formula, therefore, the bound variables are conceived to range over objects satisfying every inductive formula, including the one in question." pp.1–2

Nelson then develops a 'predicative arithmetic' as one interpretable in Robinson's theory Q. He conceives of two sets: a given infinite set of *proto*-numbers for which induction does not hold but is closed under successor, addition and multiplication; and a second set of *numbers* refined from the proto-numbers by relativisation. Nelson's central claim is that by associating predicativity with interpretability in Q, we should regard the totality of the exponential function to be an impredicative principle . There are limitations with this approach. For example, in doubting the consistency of assuming the totality of exponentiation in predicative arithmetic one is forced to assign the same doubt to the consistency of predicative arithmetic (and Robinson's arithmetic) itself by the results of Visser [60], surmised by Iwan [31]. Furthermore there have been recent results showing natural number induction to be derivable from a very weak fragment of finite-set theory (Feferman and Hellman [21]) and without appeal to finite-set induction (Ferreira [22]). We regard $EA(I; O)$ as at least carrying the predicative theme in its setting. Although precise correspondence with Nelson's notion of predicative arithmetic is not an objective within this thesis the analogy between the input/output separation and the numbers/proto-numbers is clear. Furthermore, $EA(I; O)$ does not prove the totality of exponentiation in the usual sense as input constants are not quantified and thus may be finitely limited from the outset.

We do not aim to offer model-theoretic investigations into the theories presented in this thesis and this remains an avenue for future research. We will however offer some ideas on what we envisage the models of such theories to look like by considering a model for $EA(I; O)$. Define a structure $\mathcal{M}$ with signature the non-logical symbols of $EA(I; O)$. Allow it to have a domain consisting of an infinite set $|\mathcal{O}|$ intended to interpret output variables. The non-logical symbols would be interpreted in the usual way so that $|\mathcal{O}|$ satisfies the arithmetic axioms. Crucially though $|\mathcal{O}|$ would not satisfy the induction axioms. Then join to $\mathcal{M}$ a distinguished infinite set of input constants $|\mathcal{I}| := \{i_0, i_1, \ldots, i_k, \ldots\}$. This *will* be an inductive set, intended to act as assignments to the input variables x_k. Hence the induction axiom says: if $A(a)$ is inductive in a, $A(i_k)$ holds. Then $\mathcal{M}$ would model $EA(I; O)$.

The principle objective of this thesis is to develop a hierarchy of ramified theories extending $EA(I; O)$ whose provably recursive functions correspond to higher levels of the Grzegorczyk Hierarchy. We do this through successive alternate applications of two principles, namely 'predicative induction for input constants' and 'Σ_1-closure'. The base theory for such applications has only one type of variable and resembles Robinson's arithmetic, Q. We begin by reviewing the theory $EA(I;O)$, which under our framework is viewed as the weak base theory plus predicative induction for input constants. Chapter 1 gives a lower bound on the provably recursive functions of $EA(I;O)$ and Chapter 2 provides a corresponding upper bound using a suitable infinitary theory. In these chapters we are broadly following previous work of Ostrin [46] and Ostrin and Wainer [48] but for a classical presentation of $EA(I;O)$ similar to that of Williams [66].

In Chapter 3 we define a new theory EA^1 as the weak base theory (of *one* type of variable) which incorporates a Σ_1-closure axiom. This axiom allows Σ_1 end-sequents of $EA(I;O)$ derivations to be taken as axioms where the distinction between inputs and outputs is dropped. Thus quantification now applies to inputs showing the Σ_1-closure axiom to be akin to an *internalised* ω-rule. Such a theory is (in a restricted sense) conservative over $EA(I;O)$ thus providing a setting to address the criticism that $EA(I;O)$ is not intensionally closed under substitution of provably recursive functions. We briefly discuss how the closure axiom may also be regarded as a fine graduation of, in Nelson's sense, an impredicative feature within Peano Arithmetic.

Chapter 4 begins the layering of the two principles. Firstly, a ramified theory $EA^1(I;O)$ is defined as EA^1 plus predicative induction for (new) input constants. Its provably recursive functions are shown to belong to an initial sub-class of Grzegorczyk's $\mathcal{E}^4$. Secondly, we define EA^2 as the weak base theory plus Σ_1-closure for $EA^1(I;O)$. This theory has provably recursive functions given by $\mathcal{E}^4$. Hence, beyond EA^1, the step-by-step application of the predicative induction and Σ_1-closure principles reveals both to be proper extensions of the previously defined theory.

Chapter 5 extends the layering to finitely many levels. The provably recursive functions of the single sorted theories EA^k for $0 < k \in \mathbf{N}$ are shown to be $\mathcal{E}^{k+2}$. We may then define a union, the theory $EA^{\prec\omega}$, which closes the weak base theory under an arbitrary finite number of applications of predicative induction for input constants and Σ_1-closure. This theory is equivalent to primitive recursive arithmetic, PRA. Hence we have produced a hierarchy below PRA based upon ramification of Σ_1-induction in a more implicit manner than the classical approach. In this hierarchy the two-sorted theories at each stage use predicative induction so avoid prior restrictions on the complexity of inductive formulae. According to Tait's thesis [59], the finitist would therefore accept a potentially infinite number of applications of both predicative induction and Σ_1-closure from a weak base theory such as Q. In contrast, along Nelson's lines, the *strict* finitist only accepts predicative induction.

We conclude by noting the proof-theoretic analyses employed. For each of the theories we study, the Slow-Growing Hierarchy is used to provide bounding functions for existential quantifiers in an infinitary system. Thus one would expect the proof-theoretic ordinals of the theories EA^k to be $\phi(k, 0)$ where ϕ is the Veblen hierarchy of functions. The slow-growing collapse of these ordinals provides the appropriate number-theoretic functions for the Grzegorczyk class $\mathcal{E}^{k+2}$. However, in our analysis we instead find that the layering of theories gives rise to a k-fold composition of ordinals below ε_0 which in turn yield k-many full iterations of the slow-growing functions. A more natural approach yielding the 'expected' pointwise proof-theoretic ordinals would be a first avenue for future research beyond the present thesis. There have been other different approaches to classifying PRA by ramified methods (notably the more model-theoretic work of Bellantoni [5]). However, our aim here is to develop a traditional proof-theoretic approach providing upper bounds via embeddings into carefully layered infinitary systems and stressing the slow-growing nature of the bounding functions.

Chapter 1

EA(I;O) - An Elementary Arithmetic

1.1 Preliminary Definitions

We begin by defining the arithmetic theory $EA(I;O)$ and show that within it the elementary functions are provably recursive. $EA(I;O)$ is based upon a theory developed in [18], [46], [47], [48] and [68]. The language of $EA(I;O)$ distinguishes variables into two sorts: inputs, over which induction applies; and outputs, over which quantification applies. Our particular formulation closely follows the recent variation of $EA(I;O)$ given in [63] and [66]. We work in classical logic using a Tait-style sequent calculus [58]. We include various arithmetic axioms defining the successor, predecessor, addition, recursive difference and multiplication functions. We also add axioms allowing the coding of finite sequences to simplify the task of 'bootstrapping' the theory. This will not effect the overall strength of the theory as all of these function are of sub-elementary growth rate. In general we are seeking to produce a theory analogous with classical Peano Arithmetic.

Whilst our methods simplify some matters they do incur a cost. Ostrin and Wainer [48], following Leivant's approach [34], used Kleene's equational calculus in $EA(I;O)$. That is, defining equations for all partial recursive functions are given as axioms leading to a necessary distinction between *basic* and *general* terms. Using minimal logic this allows a natural match between formal proofs and computations. These techniques draw out finer graduations of complexity in the inductive fragments of $EA(I;O)$ including a characterisation of $PTIME$. We instead choose to focus on only the full strength of the theories we introduce and shall not require such refinements at this stage.

Definition 1.1. *The language of $EA(I;O)$ has the following logical and non-logical symbols.*

- Quantifiers: existential $\exists$, and universal $\forall$.
- Propositional connectives: conjunction $\wedge$, and disjunction $\vee$.
- The binary relation symbols: equality $=$, and inequality $\neq$.
- An infinite supply of variables of two sorts: $a, b, c, d, a_0, a_1, \ldots$ for *output variables* and $x, y, z, x_0, x_1, \ldots$ for *input variables* (or rather, *uninterpreted input constants*).
- Left and right brackets, (and), for unique readability.
- The constant symbol 0.
- Unary function symbols for successor $+1$ and predecessor $\dot{-}1$, and the binary function symbols for addition $+$, recursive difference $\dot{-}$ and multiplication $\cdot$.
- Function symbols for coding of finite sequences of numbers: a binary pairing function p, a binary projection function u, and the unary function symbols: left inverse l, right inverse r, and length lh.

Definition 1.2. ***Terms*** *of $EA(I;O)$, denoted $s, t, w, t_0, t_1, \ldots$, are defined inductively.*

- The constant symbol 0, and any variable symbol a_j or x_j, is a term.
- Terms are closed under applications of any function symbol.

Where a term is constructed purely from inputs x_j (and the constant 0) we refer to it as an *input term*. To each natural number n, we associated a numeral $\overline{n}$, defined as n applications of the successor function to the constant symbol 0. However, we shall not need to distinguish between numbers n and their corresponding numerals $\overline{n}$ since it should be clear by the context which is inferred.

Definition 1.3. ***Formulae*** *of $EA(I;O)$, denoted by $A, B, C, A_0, A_1, \ldots$, are defined inductively.*

- Atomic formulae are of the form $s = t$ and $s \neq t$ for any two terms s and t.
- If A and B are formulae and a is an output variable symbol then $(A \wedge B)$, $(A \vee B)$, $\forall a(A)$ and $\exists a(A)$ are all formulae.

Where both inputs and outputs occur within a term or formulae we shall occasionally emphasise the distinction by using a semi-colon to separate them (placing inputs before the semi-colon and outputs after). We may wish to single out one or more of the free variables occurring within a term or formula by writing $t(x;a)$, $A(a)$ or $A(x)$ for example. Then by $A(t)$ we mean the formula $A(a)$ with t substituted for a throughout. The shorthand $\vec{a}$ is used to display a vector of variables $a_0, a_1, \ldots, a_j$ such as in $A(\vec{a})$. Hence $\forall \vec{a} A(\vec{a})$ means $\forall a_0 \forall a_1 \ldots \forall a_j A(a_0, a_1, \ldots, a_j)$. Since the atomic formulae occur in complimentary pairs we have no need for negation as a proper symbol of the language. Negation is defined using De Morgan's laws as follows.

Definition 1.4. *If s and t are terms and A and B are formulae then*

$$\begin{array}{rcl@{\qquad}rcl}
\neg s = t & :\equiv & s \neq t & \neg s \neq t & :\equiv & s = t \\
\neg(A \vee B) & :\equiv & (\neg A) \wedge (\neg B) & \neg(A \wedge B) & :\equiv & (\neg A) \vee (\neg B) \\
\neg \exists a(A) & :\equiv & \forall a(\neg A) & \neg \forall a(A) & :\equiv & \exists a(\neg A)
\end{array}$$

We may also use the connective symbols $\rightarrow$ and $\leftrightarrow$ by defining $A \rightarrow B :\equiv (\neg A) \vee B$ and $A \leftrightarrow B :\equiv (A \rightarrow B) \wedge (B \rightarrow A)$. We may drop some brackets by assuming that $\rightarrow$ and $\leftrightarrow$ have a wider scope than $\wedge$ and $\vee$ which in turn have a wider scope than $\neg$.

Definition 1.5. ***A Tait-Style Calculus for*** $EA(I;O)$**.**

We use a sequent calculus based upon the simplification by Tait [58] of the one-sided Gentzen-Schütte sequent calculus. In what follows, capital Greek letters $\Gamma, \Delta, \Gamma_0, \Gamma_1, \ldots$ represent finite (possibly empty) sets of formulae. Commas are used to join formulae or sets of formulae together so that Γ, A means $\Gamma \cup \{A\}$ and Γ, Δ means $\Gamma \cup \Delta$.

A *sequent* is a set of formulae which is interpreted disjunctively. That is, if $\Gamma \equiv \{A_0, A_1, \ldots, A_j\}$ then the sequent Γ is valid if and only if the disjunction $A_0 \vee A_1 \vee \ldots \vee A_j$ is. A *rule of inference* (or *rule*) in the calculus takes the form $\frac{\Gamma'}{\Gamma}$ or $\frac{\Gamma' \quad \Gamma''}{\Gamma}$ where Γ', Γ'' and Γ are sequents. We call Γ' and Γ'' the *premises* of the rule and Γ the *conclusion*. We may view axioms as rules of inference whose premise is empty.

As a consequence of using sets of formulae for sequents we do not require rules for contraction and exchange. Furthermore, we have no need for a weakening rule since we may include in the axioms all formulae of interest for a particular derivation. In each of the rules and axioms given below, the set Γ is an arbitrary set of formulae which we call the *side formulae* of the rule. The formula(e) in the premise(s) which are not side-formulae are termed the *minor formula(e)* of the rule whilst the corresponding formula in the conclusion is called the *principal formula* of the rule. When using sets of formulae we must, on account of contraction, attend to the possibility that the principal formula in each rule may already occur within the side formulae Γ in the premise(s).

The **logical axiom of excluded middle** is

(L-Ax) $\Gamma, s \neq t, s = t$ for any terms s and t.

The **logical rules** are

($\vee$) $$\frac{\Gamma, A_i}{\Gamma, A_0 \vee A_1}$$ where $i = 0$ or 1.

($\wedge$) $$\frac{\Gamma, A_0 \qquad \Gamma, A_1}{\Gamma, A_0 \wedge A_1}$$

($\forall$) $$\frac{\Gamma, A(b)}{\Gamma, \forall a A(a)}$$ where b is not free in Γ.

($\exists$) $$\frac{\Gamma, A(t)}{\Gamma, \exists a A(a)}$$ where t is the witnessing term.

The **cut rule** is

(Cut) $$\frac{\Gamma, C \qquad \Gamma, \neg C}{\Gamma}$$ where C is the cut-formula.

The **non-logical axioms** of $EA(I;O)$.

The **equality axioms** are, where t is any term:

$$\Gamma, \forall a(a = a)$$

$$\Gamma, \forall a \forall b \forall c(a = b \wedge b = c \to c = a).$$

$$\Gamma, \forall a \forall b(a = b \to t(\ldots, a, \ldots) = t(\ldots, b, \ldots)).$$

The **predicative induction** rule is, for an arbitrary set of formulae Γ

(P.Ind.) $$\frac{\Gamma, A(0) \qquad \Gamma, \neg A(a), A(a+1)}{\Gamma, A(x)}$$ where a is not free in Γ.

Usually the conclusion of such an induction rule would be $A(t(\vec{x}))$ where t is any input term. However we shall show this is derivable and it proves more convenient to use the form of induction given above. Note that since Γ is an arbitrary set of formulae, this axiom rule is equivalent to the axiom schema $\Gamma, (A(0) \wedge \forall a(A(a) \rightarrow A(a+1))) \rightarrow A(x)$ for any formulae A.

The **arithmetic axioms** are the universal closures of the following (in which we omit the mention of the side formulae Γ for clarity):

$$a + 1 \neq 0 \tag{1.1}$$

$$a + 1 = b + 1 \rightarrow a = b \tag{1.2}$$

$$a = 0 \vee (a \dot{-} 1) + 1 = a \tag{1.3}$$

$$0 \dot{-} 1 = 0 \tag{1.4}$$

$$(a + 1) \dot{-} 1 = a \tag{1.5}$$

$$a + 0 = a \tag{1.6}$$

$$a + (b + 1) = (a + b) + 1 \tag{1.7}$$

$$a \dot{-} 0 = a \tag{1.8}$$

$$a \dot{-} (b + 1) = (a \dot{-} b) \dot{-} 1 \tag{1.9}$$

$$(a + b) \dot{-} a = b \tag{1.10}$$

$$a \cdot 0 = 0 \tag{1.11}$$

$$a \cdot (b + 1) = a \cdot b + a \tag{1.12}$$

$$a + (b + c) = (a + b) + c \tag{1.13}$$

$$a + b = b + a \tag{1.14}$$

$$a \cdot (b + c) = a \cdot b + a \cdot c \tag{1.15}$$

$$(a + b) \cdot c = a \cdot c + b \cdot c \tag{1.16}$$

We include further arithmetic axioms to enable the coding of finite sequences of numbers.

$$p(a, b) \neq 0 \tag{1.17}$$

$$p(0, 0) = 1 \tag{1.18}$$

$$p(a, b + 1) = p(a, b) + a + b + 1 \tag{1.19}$$

$$p(a + 1, b) = p(a, b) + a + b + 2 \tag{1.20}$$

$$l(0) = 0 \tag{1.21}$$

$$l(p(a, b)) = a \tag{1.22}$$

$$r(0) = 0 \tag{1.23}$$

$$r(p(a, b)) = b \tag{1.24}$$

$$c = 0, c = p(l(c), r(c)) \tag{1.25}$$

$$l^0(c) = c \tag{1.26}$$

$$l^{(d+1)}(c) = l^d(l(c)) \tag{1.27}$$

$$lh(0) = 0 \tag{1.28}$$

$$lh(c + 1) = lh(l(c + 1)) + 1 \tag{1.29}$$

$$u(c, d) = r(l^{lh(c) \dot{-} (d+1)}(c)) \tag{1.30}$$

Definition 1.6. *A* derivation, deduction *or* proof-tree *in* EA^1 *is a tree* $\mathcal{D}$ *of sequents satisfying*

i. The leaves of $\mathcal{D}$ *are all either logical axioms or non-logical axioms,*

ii. Each sequent in $\mathcal{D}$*, except that at the root, is the premise of a rule of inference whose conclusion is also in* $\mathcal{D}$*.*

We write

$$EA(I;O) \vdash \Gamma$$

to mean that there is a derivation $\mathcal{D}$ *whose root node is the sequent* Γ.

Coding Finite Sequences

The paring function, p, defined as

$$p(a,b) = \frac{1}{2}(a+b)(a+b+1) + a + 1,$$

is a bijection between $\mathbf{N}^2$ and $\mathbf{N} \setminus \{0\}$. Letting 0 encode the empty sequence, we use p to append further numbers to the right of the sequence. The unary functions l and r give the left and right inverses to the function p. The function lh gives the length of the sequence. Iterating the left inverse function enables a projection function u to be defined. Henceforth we shall use the more common notation $(c)_d$ for the projection $u(c,d)$. For example, suppose we wish to find a code number c for the sequence $a_0, \ldots, a_m, \ldots, a_n$, then

$$c := p(\ldots, p(\ldots(p(p(0,a_0),a_1),\ldots)a_m)\ldots, a_n)$$

which we denote by $c = \langle a_0, \ldots, a_m, \ldots, a_n \rangle$. Hence

$$\begin{array}{rcl} l(c) & = & \langle a_0, \ldots, a_m, \ldots, a_{n-1} \rangle, \\ r(c) & = & a_n, \\ lh(c) & = & n+1, \\ (c)_m & = & a_m. \end{array}$$

Bounded Formulae

Definition 1.7. *We may conservatively extend our language to include $\leq$ and $<$ by defining*

$$\begin{aligned} a \leq b \ &:\equiv\ \exists c(a + c = b), \\ a < b \ &:\equiv\ a \leq b \wedge a \neq b. \end{aligned}$$

We use $\forall a \leq b(A(a))$ and $\exists a \leq b(A(a))$ as abbreviations for $\forall a(a \leq b \to A(a))$ and $\exists a(a \leq b \wedge A(a))$ respectively. In such cases we say that the quantifiers are *bounded*. A *bounded formula* is one in which all the quantifiers are bounded and we denote the set of such formulae by Δ_0. The rest of the arithmetic hierarchy may be given in the usual manner. Most importantly is the set of Σ_1-formulae whose members have a (possibly empty) string of existential quantifiers prefixing a bounded formula.

Proof-Tree Notation

We occasionally make use of a proof-tree notation. Solid lines are used for a rule of inference or derived rules with the rule in question noted on the right-hand side. A dotted line indicates an equivalence which has been previously defined. Common leaf abbreviations are [IH] for induction hypothesis, [L-Ax] for a logical axiom, [E-Ax] for an equality axiom and [Ax] for an arithmetic axiom. Eigenvariables will not be mentioned unless there is cause for confusion. We use exchange and contraction freely as they are build into the calculus. We also make free use of weakening by assuming that all formulae of interest have been joined to the side formulae Γ from the outset although they may not be explicitly mentioned. In doing so we must take care not to violate any eigenvariable conditions. We use informal arguments during proofs for brevity or where displayed proof-trees are too large.

1.2 Basic Results

It is necessary to derive a number of elementary results to give ourselves more liberal axioms, some useful derived rules of proof and basic properties of the inequalities $\leq$ and $<$. The proofs of the claims in the following two lemmas are straightforward so we only provide brief details. We include a fuller exposition in Appendix A.

Lemma 1.8. *Letting* Γ *be an arbitrary set of side formulae,* A, A_0, A_1 *be any formulae,* s, t, w *be any terms, and* $i = 0$ *or* 1*, we have:*

1. The Generalized law of excluded middle. $EA(I;O) \vdash \Gamma, \neg A, A$.

2. Conjunction inversion. $EA(I;O) \vdash \Gamma, A_0 \wedge A_1 \;\Rightarrow\; EA(I;O) \vdash \Gamma, A_i$

3. Disjunction inversion. $EA(I;O) \vdash \Gamma, A_0 \vee A_1 \;\Rightarrow\; EA(I;O) \vdash \Gamma, A_0, A_1$.

4. Universal quantifier inversion. $EA(I;O) \vdash \Gamma, \forall a A(a) \;\Rightarrow\; EA(I;O) \vdash \Gamma, A(t)$.

5. Symmetry of equality. $EA(I;O) \vdash \Gamma, s \neq t, t = s$.

6. Transitivity of equality. $EA(I;O) \vdash \Gamma, s \neq t, t \neq w, s = w$.

7. Generalized law of equality. $EA(I;O) \vdash \Gamma, s \neq t, \neg A(s), A(t)$.

8. Substitution. $EA(I;O) \vdash \Gamma, s = t \;\&\; EA(I;O) \vdash \Gamma, A(s) \;\Rightarrow\; EA(I;O) \vdash \Gamma, A(t)$.

9. Cases. $EA(I;O) \vdash \Gamma, A(0) \;\&\; EA(I;O) \vdash \Gamma, A(a+1) \;\Rightarrow\; EA(I;O) \vdash \Gamma, A(t)$, *provided* a *is not free in* Γ.

Proofs.

Part 1 uses induction over the build up of the formula A. Henceforth we use this result as an axiom still denoted by (L-Ax). The inversions are all given similarly, for example part 4 results from

$$\dfrac{\dfrac{\overset{[\text{L-Ax}]}{\Gamma, \neg A(t), A(t)}}{\begin{array}{c}\Gamma, \exists a(\neg A(a)), A(t)\\ \hdashline \Gamma, \neg\forall a(A(a)), A(t)\end{array}}\,(\exists) \qquad \overset{[\text{Assumption}]}{\Gamma, \forall a A(a)}}{\Gamma, A(t).}\,(\text{Cut})$$

We use inversions in proof-trees as derived rules in their own right with the notations ($\wedge$-inv), ($\vee$- inv) and ($\forall$-inv) respectively. They may be used to give more liberal equality and arithmetic axioms such as $\Gamma, s \neq t, t \neq w, w = s$. We still denote such sequents as [E-Ax] or [Ax]. Parts 5 and 6 are simple derivations from equality axioms and 7 uses induction over the build up of A. We shall now use the leaf abbreviation [E-Ax] to also refer to instances of parts 5, 6 and 7. From part 7, part 8 follows. Part 9 results from the inclusion of arithmetic axiom $a = 0 \vee (a \mathbin{\dot{-}} 1) + 1 = a$ and we give the proof here:

Firstly we have

$$\dfrac{\overset{[\text{Assumption}]}{\Gamma, A(0)} \qquad \overset{[\text{Part 7}]}{\Gamma, t \neq 0, \neg A(0), A(t)}}{\Gamma, t \neq 0, A(t).}\,(\text{Cut})$$

Secondly

$$\dfrac{\dfrac{\overset{[\text{Assumption}]}{\Gamma, A(a+1)} \qquad \overset{[\text{Part 7}]}{\Gamma, a+1 \neq t, \neg A(a+1), A(t)}}{\Gamma, a+1 \neq t, A(t)}\,(\text{Cut})}{\Gamma, \forall a(a+1 \neq t), A(t).}\,(\forall)$$

Forming the conjunction of the two derivations above we obtain

$$\dfrac{\Gamma, t \neq 0 \wedge \forall a(a+1 \neq t), A(t) \qquad \dfrac{\dfrac{\dfrac{\overset{[\text{Ax (1.3)}]}{\Gamma, t \neq 0 \vee (t \mathbin{\dot{-}} 1) + 1 = t}}{\Gamma, t \neq 0, (t \mathbin{\dot{-}} 1) + 1 = t}\,(\vee\text{-inv})}{\Gamma, t \neq 0, \exists a(a+1 = t)}\,(\exists)}{\Gamma, t \neq 0 \vee \exists a(a+1 = t).}\,(\vee)}{\Gamma, A(t).}\,(\text{Cut})$$

We also now use parts 8 and 9 as derived rules within proof-trees adopting the notation (Sub.) or (Cases) respectively.

$\square$

From this point forward we may reduce the clutter by omitting the mention of side formulae Γ although they are assumed to be present.

Lemma 1.9. *In $EA(I;O)$ we may derive the universal closures of:*

1. $a \leq a$.

2. $a \leq 0 \rightarrow a = 0$.

3. $a \leq b+1 \rightarrow a \leq b \vee a = b+1$.

4. $a \leq b \rightarrow a \leq b+1$.

5. $a+1 \leq b \rightarrow a \leq b$.

6. $a \leq b \rightarrow a+1 \leq b+1$.

7. $a \leq b \wedge a' \leq b' \rightarrow a+a' \leq b+b'$.

8. $a \leq b \wedge a' \leq b' \rightarrow a \cdot a' \leq b \cdot b'$.

9. $\neg 0 < a$.

10. $a < a+1$.

11. $a < b \rightarrow a+1 \leq b$.

12. $a < b \rightarrow a+1 < b+1$.

13. $a < b \rightarrow a < b+1$.

14. $a < b+1 \rightarrow a < b \vee a = b$.

Proofs.

These results are given by straightforward derivations from the arithmetic axioms and may be found in Appendix A.

$\square$

Definition 1.10. *A formula A is said to be* ***progressive*** *in variable a, written $Prog_a A(a)$, if and only if it is inductive. That is,*

$$Prog_a A(a) :\equiv A(0) \wedge \forall a(A(a) \rightarrow A(a+1)).$$

Thus our predicative induction axiom rule is equivalent to the axiom schema

$$EA(I;O) \vdash \Gamma, Prog_a A(a) \rightarrow A(x)$$

for any formulae A.

In $EA(I;O)$, once an input variable x is introduced it remains free since quantification only applies to outputs. Hence we may see these variables as being *input constants*. This feature is crucial in restricting the strength of the theory. However, it is also the source of the key drawback of working within a theory such as $EA(I;O)$: there is no *natural* method for substitution on inputs within the theory. This is a criticism we address directly in Chapter 3. For now we shall present some of the ways in which variations on the predicative induction rule may be derived. We follow the techniques in Ostrin and Wainer [48] and Williams [66]. These will include 'input bounded output induction' and 'induction up to any polynomial input term'. We refer the reader to Wirz [68] for a generalized analysis of extensions to the predicative induction rule including an input substitution rule.

Lemma 1.11. *For any formula $A(a)$*

$$EA(I;O) \vdash Prog_a A(a) \rightarrow Prog_b \forall a(a \leq b \rightarrow A(a)).$$

Proof.

Firstly we have

$$\dfrac{\dfrac{\overset{\text{[L-Ax]}}{\neg Prog_a A(a), Prog_a A(a)}}{\neg Prog_a A(a), A(0)}\ (\wedge\text{-inv}) \qquad \overset{\text{[E-Ax]}}{a \neq 0, \neg A(0), A(a)}}{\neg Prog_a A(a), a \neq 0, A(a)}\ \text{(Cut)}$$

from which we deduce

$$\dfrac{\dfrac{\dfrac{\neg Prog_a A(a), a \neq 0, A(a) \qquad \overset{\text{[Lemma 1.9 part 2]}}{\neg a \leq 0, a = 0}}{\neg Prog_a A(a), \neg a \leq 0, A(a)}\ \text{(Cut)}}{\neg Prog_a A(a), \neg a \leq 0 \vee A(a)}\ (\vee)}{\neg Prog_a A(a), \forall a(\neg a \leq 0 \vee A(a)).}\ (\forall)$$

This sequent contains the first conjunct of $Prog_b \forall a(a \leq b \to A(a))$, the base case of progressiveness.

To find the second conjunct let us put $B(b) :\equiv \forall a(a \leq b \to A(a))$. We start with

$$\dfrac{\dfrac{\dfrac{\overset{\text{[L-Ax]}}{\neg Prog_a A(a), Prog_a A(a)}}{\neg Prog_a A(a), \forall a(\neg A(a) \vee A(a+1))}\ (\wedge\text{-inv})}{\neg Prog_a A(a), \neg A(b) \vee A(b+1)}\ (\forall\text{-inv})}{\neg Prog_a A(a), \neg A(b), A(b+1)}\ (\vee\text{-inv})$$

and

$$\dfrac{\dfrac{\dfrac{\overset{\text{[L-Ax]}}{\neg B(b), B(b)}}{\neg B(b), \neg b \leq b \vee A(b)}\ (\forall\text{-inv})}{\neg B(b), \neg b \leq b, A(b)}\ (\vee\text{-inv}) \qquad \overset{\text{[Lemma 1.9 part 1]}}{b \leq b}}{\neg B(b), A(b).}\ \text{(Cut)}$$

which combined via a cut yield $\neg Prog_a A(a), \neg B(b), A(b+1)$. By one further cut with the equality axiom $a \neq b+1, \neg A(b+1), A(a)$ we continue with

$$\dfrac{\neg Prog_a A(a), \neg B(b), a \neq b+1, A(a) \qquad \overset{\text{[Lemma 1.9 part 3]}}{\neg a \leq b+1, a \leq b, a = b+1}}{\neg Prog_a A(a), \neg B(b), \neg a \leq b+1, a \leq b, A(a)}\ \text{(Cut)}$$

and then

$$\dfrac{\dfrac{\dfrac{\neg Prog_a A(a), \neg B(b), \neg a \leq b+1, a \leq b, A(a) \qquad \dfrac{\dfrac{\overset{[\text{L-Ax}]}{\neg B(b), B(b)}}{\neg B(b), \neg a \leq b \vee A(a)}\,(\forall\text{-inv})}{\neg B(b), \neg a \leq b, A(a)}\,(\vee\text{-inv})}{\neg Prog_a A(a), \neg B(b), \neg a \leq b+1, A(a)}\,(\text{Cut})}{\neg Prog_a A(a), \neg B(b), \neg a \leq b+1 \vee A(a)}\,(\vee)}{\neg Prog_a A(a), \neg B(b), B(b+1).}\,(\forall)$$

This provides the second conjunct of $Prog_b \forall a(a \leq b \to A(a))$ by disjunction and universal quantification over b. Hence

$$EA(I;O) \vdash Prog_a A(a) \to Prog_b \forall a(a \leq b \to A(a)).$$

□

Lemma 1.12. *For any formula* $A(a)$

$$EA(I;O) \vdash Prog_a A(a) \to Prog_b \forall a(A(a) \to A(a+b)).$$

Proof.

We show the base case for progressiveness in b of $\forall a(A(a) \to A(a+b))$ as follows:

$$\dfrac{\dfrac{\dfrac{\overset{[\text{L-Ax}]}{\neg Prog_a A(a), \neg A(a), A(a)} \qquad \overset{[\text{Ax (1.6)}]}{a+0=a}}{\neg Prog_a A(a), \neg A(a), A(a+0)}\,(\text{Sub.})}{\neg Prog_a A(a), \neg A(a) \vee A(a+0)}\,(\vee)}{\neg Prog_a A(a), \forall a(\neg A(a) \vee A(a+0))}\,(\forall)$$

The inductive step begins with

$$\dfrac{\dfrac{\dfrac{\overset{[\text{L-Ax}]}{\neg Prog_a A(a), Prog_a A(a)}}{\neg Prog_a A(a), \forall a(\neg A(a) \vee A(a+1))}\,(\wedge\text{-inv})}{\neg Prog_a A(a), \neg A(a+b) \vee A((a+b)+1)}\,(\forall\text{-inv})}{\neg Prog_a A(a), \neg A(a+b), A((a+b)+1).}\,(\vee\text{-inv})$$

A use of substitution with the arithmetic axiom $a + (b + 1) = (a + b) + 1$ yields

$$\neg Prog_a A(a), \neg A(a + b), A(a + (b + 1)). \tag{1.31}$$

Now, putting $B(b) :\equiv \forall a(\neg A(a) \lor A(a + b))$, we have

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\overset{[\text{L-Ax}]}{\neg B(b), B(b)}}{\neg B(b), \neg A(a) \lor A(a + b)}\,(\forall\text{-inv})}{\neg B(b), \neg A(a), A(a + b)}\,(\lor\text{-inv}) \qquad [(1.31)]}{\neg Prog_a A(a), \neg B(b), \neg A(a), A(a + (b + 1))}\,(\text{Cut})}{\neg Prog_a A(a), \neg B(b), \neg A(a) \lor A(a + (b + 1))}\,(\lor)}{\neg Prog_a A(a), \neg B(b), B(b + 1).}\,(\forall)$$

Hence by disjunction, universal quantification and finally conjunction with the base case we find

$$EA(I;O) \vdash \neg Prog_a A(a), Prog_b \forall a(A(a) \to A(a + b)).$$

$\square$

Lemma 1.13. *Let $t(\vec{x})$ be any polynomial term on inputs. That is t only uses $+1, +, \cdot$ on inputs x_k or the constant 0. Then for any formula $A(a)$*

$$EA(I;O) \vdash Prog_a A(a) \to A(t(\vec{x})).$$

Proof.

We use induction over the build up of the term t.

1. If t is the constant 0 then $Prog_a A(a)$ entails $A(0)$ by conjunction inversion. When t is x the result follows from the predicative induction rule.

2. Assume the result holds for t_0 where $t := t_0 + 1$. We have

$$\dfrac{\dfrac{\dfrac{\dfrac{\overset{[\text{L-Ax}]}{\neg Prog_a A(a), Prog_a A(a)}}{\neg Prog_a A(a), \forall a(\neg A(a) \lor A(a + 1))}\,(\land\text{-inv})}{\neg Prog_a A(a), \neg A(t_0) \lor A(t_0 + 1)}\,(\forall\text{-inv})}{\neg Prog_a A(a), \neg A(t_0), A(t_0 + 1)}\,(\lor\text{-inv}) \qquad \overset{[\text{IH}]}{\neg Prog_a A(a), A(t_0)}}{\neg Prog_a A(a), A(t_0 + 1).}\,(\text{Cut})$$

3. Now let $t := t_0 + t_1$ and assume the result holds for the sub-terms. We make use of the Lemma 1.12 letting $B(b) :\equiv \forall a(\neg A(a) \vee A(a+b))$ and the induction hypothesis for t_1 on $B(b)$ as follows:

$$\frac{\overset{\text{Lemma 1.12}}{\neg Prog_a A(a), Prog_b B(b)} \qquad \overset{[\text{IH}]}{\neg Prog_b B(b), B(t_1)}}{\neg Prog_a A(a), B(t_1).}\ (\text{Cut})$$

Inverting the universal quantifier in $B(t_1)$ at the term t_0 and using the induction hypothesis on t_0 gives

$$\frac{\neg Prog_a A(a), \neg A(t_0), A(t_0 + t_1) \qquad \overset{[\text{IH}]}{\neg Prog_a A(a), A(t_0)}}{\neg Prog_a A(a), A(t_0 + t_1).}\ (\text{Cut})$$

4. Finally let $t := t_0 \cdot t_1$ with the result holding for t_0 and t_1. Then define $B(b)$ as above. Firstly

$$\begin{array}{c} \dfrac{\overset{\text{Lemma 1.12}}{\neg Prog_a A(a), Prog_b B(b)} \qquad \overset{[\text{IH}]}{\neg Prog_b B(b), B(t_0)}}{\neg Prog_a A(a), B(t_0)}\ (\text{Cut}) \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \dfrac{\neg Prog_a A(b), \forall a(\neg A(a) \vee A(a + t_0))}{\neg Prog_a A(a), \neg A(t_0 \cdot b) \vee A((t_0 \cdot b) + t_0)).}\ (\forall\text{-inv}) \end{array}$$

Then by the substitution $t_0 \cdot (b+1) = (t_0 \cdot b) + t_0$ and universal quantification we leave

$$\neg Prog_a A(a), \forall b(\neg A(t_0 \cdot b) \vee A(t_0 \cdot (b+1))).$$

Also, since $Prog_a A(a)$ entails $A(0)$, we have $\neg Prog_a A(a), A(t_0 \cdot 0)$. Thus by conjunction

$$\neg Prog_a A(a), Prog_b A(t_0 \cdot b).$$

Finally we make use of the induction hypothesis for t_1 by an application to the formula $A(t_0 \cdot b)$.

$$\frac{\neg Prog_a A(a), Prog_b A(t_0 \cdot b) \qquad \overset{[\text{IH}]}{\neg Prog_b A(t_0 \cdot b), A(t_0 \cdot t_1)}}{\neg Prog_a A(a), A(t_0 \cdot t_1)}\ (\text{Cut})$$

Hence we see

$$EA(I;O) \vdash Prog_a A(a) \rightarrow A(t(\vec{x}))$$

provided t is a polynomial input term.

$\square$

Corollary 1.14. *Given any polynomial term $t(\vec{x})$ and any formula $A(a)$*

$$EA(I;O) \vdash Prog_a A(a) \rightarrow \forall a \leq t(\vec{x}) A(a).$$

Proof.

We use the previous result in conjunction with Lemma 1.11.

$\square$

This form of input bounded predicative induction may be used to establish that for *any* input term $s(\vec{x})$,

$$EA(I;O) \vdash Prog_a A(a) \rightarrow A(s(\vec{x}))$$

by proving any such term will always be bounded by some polynomial $t(\vec{x})$. However, for our purposes it will suffice to know that predicative induction up to any polynomial input term holds.

1.3 Exponentiation in $EA(I;O)$

The principal result of this section is that the exponential function is provably recursive in $EA(I;O)$. We must first begin with a definition of what it means for a function to be provably recursive in our context.

Definition 1.15. *A function f is said to be **provably recursive** in the theory $EA(I;O)$ if its graph can be defined by a Σ_1-formula (that is there is a Σ_1-formula $\exists c C_f(\vec{a}, b, c)$,* *where C_f is Δ_0, and for which $f(\vec{n}) = m$ if and only if $\exists c(C_f(\vec{n}, m, c))$ is true) such that for inputs $\vec{x}$*

$$EA(I;O) \vdash \exists b \exists c(C_f(\vec{x}; b, c))$$

$$EA(I;O) \vdash \forall b \forall b'(\exists c(C_f(\vec{x}; b, c)) \wedge \exists c(C_f(\vec{x}; b', c)) \rightarrow b = b').$$

We call these requirements the existence and the uniqueness conditions for f. We denote the class of functions provably recursive in $EA(I;O)$ by $ProvRec(EA(I;O))$.

We call $C_f(\vec{x}; b, c)$ the *computational formula* for f. The idea is that c is a code for the sequence computing successive values of $f(\vec{x})$. The existence condition is the principal requirement for a function to be provably recursive as the uniqueness condition is usually a corollary of existence. Note that normally such a definition would require existence and uniqueness conditions to hold for every x. However we cannot prove a suitable Π_2-formula since inputs remain free in $EA(I;O)$. This subtle distinction shall be expanded on in Chapter 3.

Definition 1.16. *We define a computational formula $E(a, b, c)$ for the exponential function $2^a = b$ as follows:*

$$E(a, b, c) :\equiv lh(c) = a + 1 \wedge (c)_0 = 1 \wedge r(c) = b \wedge \forall d < a((c)_{d+1} = (c)_d + (c)_d).$$

This definition determines a sequence code $c :\equiv \langle 1, 2, 4, \ldots, 2^a \rangle$. Hence $2^n = m$ if and only if $\exists c E(n, m, c)$ is true.

In order to prove $\exists b \exists c E(x; b, c)$ and $\forall b \forall b'(\exists c E(x; b, c) \wedge \exists c E(x; b', c)) \rightarrow b = b')$ we first need a preliminary lemma regarding sequence codes.

Lemma 1.17.

$$EA(I;O) \vdash \forall c \forall b \forall d < lh(c)((c)_d = (p(c, b))_d).$$

Proof.

Using axiom (1.10) we are able to show

$$(((d + 1) + a) \dot{-} (d + 1)) + 1 = a + 1 = ((d + 1) + (a + 1)) \dot{-} (d + 1).$$

Hence by equality and associativity of addition

$$(d+1)+a \neq lh(c), (lh(c)\dot{-}(d+1))+1 = (lh(c)+1)\dot{-}(d+1).$$

By universally quantifying a this is equivalent to

$$d+1 \leq lh(c) \rightarrow (lh(c)\dot{-}(d+1))+1 = (lh(c)+1)\dot{-}(d+1).$$

We know by (1.22) that $l(p(c,b)) = c$. Furthermore $p(c,b)$ is non-zero by (1.17). Hence by (1.29) we have $lh(p(c,b)) = lh(c)+1$. Substituting in we have

$$d+1 \leq lh(c) \rightarrow (lh(c)\dot{-}(d+1))+1 = lh(p(c,b))\dot{-}(d+1).$$

By a cut with an equality axiom

$$d+1 \leq lh(c) \rightarrow r(l^{(lh(c)\dot{-}(d+1))+1}(p(c,b))) = r(l^{lh(p(c,b))\dot{-}(d+1)}(p(c,b))).$$

Then, using $l(p(c,b)) = c$ and with axiom (1.27),

$$d+1 \leq lh(c) \rightarrow r(l^{(lh(c)\dot{-}(d+1))}(c)) = r(l^{lh(p(c,b))\dot{-}(d+1)}(p(c,b)))$$

which is, by the defining axiom for the projection function (1.30),

$$d+1 \leq lh(c) \rightarrow (c)_d = (p(c,b))_d.$$

A final cut with part 11 of Lemma 1.9 leaves

$$d < lh(c) \rightarrow (c)_d = (p(c,b))_d.$$

Applying universal quantifiers over d, then b and finally c will complete the derivation.

□

Lemma 1.18.

$$EA(I;O) \vdash Prog_a(\exists b \exists c E(a,b,c)).$$

Proof.

We shall make extensive use of the coding axioms provided on page 14 and argue inside $EA(I;O)$ using an informal natural deduction style. Recall the definition of $E(a,b,c)$ is

$$E(a,b,c) :\equiv lh(c) = a + 1 \wedge (c)_0 = 1 \wedge r(c) = b \wedge \forall d < a((c)_{d+1} = (c)_d + (c)_d).$$

We start with the base case of progressiveness. We look to prove each of the four conjuncts in $E(0,1,p(0,1))$ from which existential quantifications will leave $\exists b \exists c E(0,b,c)$.

i. By axioms (1.18) and (1.19) we find $p(0,1) = 1 + 1$. Hence axioms (1.22), (1.28) and (1.29) give $lh(p(0,1)) = lh(l(p(0,1)) + 1 = 0 + 1$.

ii. Axiom (1.24) ensures $r(p(0,1)) = 1$ and by (1.26) $l^0(p(0,1)) = p(0,1)$. Since $lh(p(0,1)) = 0 + 1$, we have by (1.5), $lh(p(0,1)) \dot{-} 1 = 0$. Hence by (1.30) we see $(p(0,1))_0 = r(l^0(p(0,1))) = r(p(0,1)) = 1$.

iii. $r(p(0,1)) = 1$ is an instance of axiom (1.24).

iv. Lastly by Lemma 1.9 part 9, $\neg d < 0$. Hence the fourth conjunct follows by weakening.

Now we move on to the inductive step. Assume that we are given b and c such that $E(a,b,c)$ holds. We show the conjuncts in $E(a+1, b+b, p(c, b+b))$ hold.

i. Firstly $p(c, b+b)$ is non-zero by (1.17). By axioms (1.22) and (1.29) we find $lh(p(c,b+b)) = lh(l(p(c,b+b))) + 1 = lh(c) + 1$. Then $lh(p(c,b+b)) = (a+1) + 1$ by our assumption for c.

ii. Under the assumption for c we know $(c)_0 = 1$ and $lh(c) = a + 1$ hence $r(l^a(c)) = 1$ by axiom (1.30). Since $l(p(c,b+b)) = c$ by (1.22) and $lh(p(c,b+b)) = (a+1)+1$ by part i, it follows that using (1.5), (1.27) and (1.30)

$$(p(c,b+b))_0 = r(l^{a+1}(p(c,b+b))) = r(l^a(l(p(c,b+b))) = r(l^a(c)) = 1.$$

iii. $r(p(c,b+b)) = b + b$ is an instance of axiom (1.24).

iv. If $d < a + 1$ then $d = a$ or $d < a$ by part 14 of 1.9. Firstly assume that $d = a$. Then since $((a+1)+1)\dot{-}((a+1)+1) = 0$ by (1.6) and (1.10) and $lh(p(c, b+b)) = (a+1)+1$ by part i, axioms (1.24), (1.26), (1.30) imply

$$(p(c, b + b))_{a+1} = r(l^0(p(c, b + b))) = r(p(c, b + b)) = b + b.$$

By the assumption $b = r(c)$ in $E(a, b, c)$ we find, using (1.10), (1.22), (1.27) and (1.30),

$$b = r(l(p(c, b + b)) = r(l^1(p(c, b + b)) = (p(c, b + b))_a$$

Hence

$$(p(c, b + b))_{a+1} = b + b = (p(c, b + b))_a + (p(c, b + b))_a. \tag{1.32}$$

Now assume that $d < a$. By the preceding Lemma 1.17, since $lh(c) = a + 1$ we have $\forall d < a + 1((c)_d = (p(c, b + b))_d)$. Inverting at d and using part 13 of Lemma 1.9 yields $(c)_d = (p(c, b + b))_d$. Whilst inverting at $d + 1$ and using part 12 of Lemma 1.9 gives $(c)_{d+1} = (p(c, b + b))_{d+1}$. Recall that since $d < a$ the assumption $E(a, b, c)$ informs us that $(c)_{d+1} = (c)_d + (c)_d$. Bringing these observations together we find

$$(p(c, b + b))_{d+1} = (c)_{d+1} = (c)_d + (c)_d = (p(c, b + b))_d + (p(c, b + b))_d. \tag{1.33}$$

Hence from (1.32) and (1.33)

$$d < a + 1 \rightarrow (p(c, b + b))_{d+1} = (p(c, b + b))_d + (p(c, b + b))_d$$

and universal quantification gives the final conjunct.

Taking the conjunction of i.-iv. above we have $E(a + 1, b + b, p(c, b + b))$. Hence we existentially quantify with witnesses $p(c, b + b)$ and $b + b$ to leave $\exists b \exists c E(a + 1, b, c)$. That is we have shown

$$\exists b \exists c E(a, b, c) \rightarrow \exists b \exists c E(a + 1, b, c).$$

Finally bringing into conjunction the base case and the universally quantified inductive step we conclude

$$EA(I;O) \vdash \exists b \exists c E(0, b, c) \wedge \forall a(\exists b \exists c E(a, b, c) \rightarrow \exists b \exists c E(a + 1, b, c)).$$

□

Lemma 1.19.

$$EA(I;O) \vdash Prog_a(\forall b \forall b'(\exists c E(a,b,c) \wedge \exists c E(a,b',c) \rightarrow b = b')).$$

Proof.

We follow the style of the previous proof.

For the base of progressiveness we prove $\exists c E(0,b,c) \wedge \exists c E(0,b',c) \rightarrow b = b'$. Firstly assume that we are given a c such that $E(0,b,c)$ and secondly that we are given a c' such that $E(0,b',c')$. From the first assumption, since $lh(c) = 1$, by equality with axiom (1.5) and (1.30) we find $(c)_0 = r(l^0(c))$. Using (1.26) this is $(c)_0 = r(c)$. Appealing to $E(0,b,c)$ again we know $(c)_0 = 1$ and $r(c) = b$. Hence by equality $b = 1$. By the same argument applied to the second assumption we find $b' = 1$. Hence by equality we are left with $b = b'$.

To verify the inductive step we firstly assume

$$\forall b \forall b'(\exists c E(a,b,c) \wedge \exists c E(a,b',c) \rightarrow b = b') \tag{1.34}$$

and then secondly assume

$$\exists c E(a+1,b,c) \wedge \exists c E(a+1,b',c) \tag{1.35}$$

in order to deduce that $b = b'$.

We shall start with the assumption that there exists a c such that $E(a+1,b,c)$. We recall that

$$E(a+1,b,c) :\equiv lh(c) = (a+1)+1 \wedge (c)_0 = 1 \wedge r(c) = b \wedge \forall d < a+1((c)_{d+1} = (c)_d + (c)_d).$$

Our aim is to prove that $E(a, r(l(c)), l(c))$ holds. There are four conjuncts to check.

i. Under the assumption for c we have $lh(c) = (a+1)+1$ thus c is non-zero. Hence $lh(l(c)) = a+1$ by axiom (1.29).

ii. Since $lh(c) = (a+1)+1$ by the assumption for c, we use equality and substitution on axioms (1.5), (1.27), (1.29) and (1.30) to deduce

$$(l(c))_0 = r(l^{lh(l(c)) \dot{-} 1}(l(c))) = r(l^a(l(c))) = r(l^{a+1}(c)) = r(l^{lh(c) \dot{-} 1}(c)) = (c)_0.$$

Hence, as $(c)_0 = 1$ by the assumption for c, we deduce $(l(c))_0 = 1$.

iii. $r(l(c)) = r(l(c))$ is an axiom.

iv. The assumption for c gives $\forall d < a+1((c)_{d+1} = (c)_d + (c)_d)$. As c is non-zero $c = p(l(c), r(c))$ by axiom (1.25). We shall show $\forall d < a((l(c))_{d+1} = (l(c))_d + (l(c))_d)$.

Using appropriate inversions Lemma 1.17 reads $\forall d < lh(l(c))((l(c))_d = (p(l(c), r(c)))_d)$. Then since $c = p(l(c), r(c))$ and $lh(l(c)) = a+1$ by part i, this is

$$\forall d < a+1((l(c))_d = (c)_d).$$

Recall that parts 12 and 13 of Lemma 1.9 show that $d < a$ implies both $d+1 < a+1$ and $d < a+1$. Hence by inverting at $d+1$ and then d, if we assume $d < a$ we have by cuts both $(l(c))_d = (c)_d$ and $(l(c))_{d+1} = (c)_{d+1}$. Likewise from the assumption for c if $d < a$ then $(c)_{d+1} = (c)_d + (c)_d$.

Together this gives, if $d < a$ then

$$(l(c))_{d+1} = (c)_{d+1} = (c)_d + (c)_d = (l(c))_d + (l(c))_d.$$

That is $d < a \to (l(c))_{d+1} = (l(c))_d + (l(c))_d$. Universal quantification leaves what we require for the fourth conjunct of $E(a, r(l(c)), l(c))$.

We have now established, by parts i.-iv. above, that $E(a, r(l(c)), l(c))$ follows from the first conjunct of assumption (1.35). By the same argument applied to the second conjunct of assumption (1.35) we may also deduce $E(a, r(l(c')), l(c'))$. Existential quantifications give

$$\exists c E(a, r(l(c)), c) \wedge \exists c E(a, r(l(c')), c). \tag{1.36}$$

Now we look to use (1.34). By inverting the universal quantifiers at $r(l(c))$ and $r(l(c'))$ respectively we may cut with (1.36) above to conclude $r(l(c)) = r(l(c'))$. We shall use this to show $b = b'$.

We know $\forall d < a + 1((c)_{d+1} = (c)_d + (c)_d)$ and $\forall d < a + 1((c')_{d+1} = (c')_d + (c')_d)$ from assumption (1.35). As $a < a + 1$ is provable by Lemma 1.9 part 10, inverting at a allows us to conclude $(c)_{a+1} = (c)_a + (c)_a$ and $(c')_{a+1} = (c')_a + (c')_a$. However as $r(l(c)) = r(l(c'))$ it is easy to show $(c)_a = (c')_a$ using axioms (1.10) and (1.30) since $lh(c) = lh(c') = (a + 1) + 1$. We now deduce by equality $(c)_{a+1} = (c')_{a+1}$. But $(c)_{a+1}$ is $r(c)$ and $(c')_{a+1}$ is $r(c')$ by axiom (1.30), hence $r(c) = r(c')$. From assumption (1.35) we know $r(c) = b$ and $r(c') = b'$. Therefore $b = b'$.

Having deduced $b = b'$ from assumptions (1.34) and (1.35) we know

$$\begin{aligned}&\forall b \forall b' (\exists c E(a, b, c) \wedge \exists c E(a, b', c) \to b = b')\\ &\qquad \to \forall b \forall b' (\exists c E(a + 1, b, c) \wedge \exists c E(a + 1, b', c) \to b = b').\end{aligned}$$

Universal quantification over a and conjunction with the base case yields

$$Prog_a(\forall b \forall b' (\exists c E(a, b, c) \wedge \exists c E(a, b', c) \to b = b')).$$

□

Corollary 1.20. *The function* 2^x *is provably recursive in* $EA(I;O)$.

Proof.

Applying predicative induction to each of the last two lemmas, 1.18 and 1.19, satisfies the requirements of Definition 1.15 for the function 2^x.

□

1.4 The Elementary Functions Are Provably Recursive in $EA(I;O)$

The main result of this chapter is that the Kalmár-Csillag elementary functions, $\mathcal{E}^3$ in the Grzegorczyk Hierarchy, are provably recursive in $EA(I;O)$. We follow the approach laid out in [48]. We further extend the predicative induction rule to apply up to any finite iterate of the exponential function on inputs. The argument used dates back to Gentzen ([24],[25]) but with numbers in place of ordinals. We then use the fact that any elementary function is computable in a number of steps bounded by a finite iterate of the exponential function. Alternative approaches to reaching this result are given in [46] and [66] based on different characterizations of the elementary functions.

Definition 1.21. *Let $2_k(x)$ for a fixed $k \in \mathrm{N}$ denote the k-times iterate of 2^x. That is $2_0(x) := x$ and $2_{k+1}(x) := 2^{2_k(x)}$. Then for any formula $A(a)$ we define*

$$A(2^a) :\equiv \exists b(\exists c E(a,b,c) \wedge A(b)), \qquad A(2_{k+1}(x)) :\equiv A(2^{2_k(x)})$$

and

$$A'(d) :\equiv \forall a(A(a) \rightarrow \exists b(\exists c E(d,b,c) \wedge A(a+b))).$$

Lemma 1.22.

$$EA(I;O) \vdash Prog_a A(a) \rightarrow Prog_d A'(d).$$

Proof.

We argue informally in a natural deduction style. Assume $Prog_a A(a)$.

i. We first show $A(a) \rightarrow \exists b(\exists c E(0,b,c) \wedge A(a+b))$ from which $A'(0)$ follows by universal quantification over a.

Assume $A(a)$. Then by $Prog_a A(a)$ we have $A(a+1)$. From the proof of Lemma 1.18 we know $\exists c E(0,1,c)$ is derivable. Hence by conjunction and existential quantification $\exists b(\exists c E(0,b,c) \wedge A(a+b))$.

ii. Now assume $A'(d)$. We show $A(a) \to \exists b(\exists c E(d+1,b,c) \wedge A(a+b))$ which entails $A'(d+1)$ by universal quantification over a. We shall require two appeals to $A'(d)$.

Assume $A(a)$. Then by inverting the universal quantifier in $A'(d)$ at a, we know there exists a b such that $\exists c E(d,b,c) \wedge A(a+b)$. Firstly, for this b we may again invert the universal quantifier in $A'(d)$ at $a+b$ so that from $A(a+b)$ we find $A(a+b+b)$. Secondly, within the proof of Lemma 1.18 we showed, given a b and a c such that $E(d,b,c)$, it follows that $E(d+1,b+b,p(c,b+b))$. Hence $\exists c E(d+1,b+b,c)$.

Forming the conjunction $\exists c E(d+1,b+b,c) \wedge A(a+b+b)$ we may existentially quantify at the witness $b+b$ to leave $\exists b(\exists c E(d+1,b,c) \wedge A(a+b))$.

□

Lemma 1.23. *Given any polynomial term* $t(\vec{x})$ *and any formula* $A(a)$*, for each* $k \in \mathrm{N}$

$$EA(I;O) \vdash Prog_a A(a) \to A(2_k(t(\vec{x})))$$

Proof.

We use a meta-induction on k. If $k := 0$ the result is given by Lemma 1.13.

Now assume the result holds for k. Applying the induction hypothesis to the formula $A'(d)$ we have $Prog_d A'(d) \to A'(2_k(t(\vec{x})))$. By a cut with the previous lemma we deduce $Prog_a A(a) \to A'(2_k(t(\vec{x})))$. Recalling Definition 1.21, this is

$$Prog_a A(a) \to \forall a(A(a) \to \exists b(\exists c E(2_k(t(\vec{x})), b, c) \wedge A(a+b))).$$

We invert the universally quantified a at 0 and then cut $A(0)$ by appealing to $Prog_a A(a)$. This leaves

$$Prog_a A(a) \to \exists b(\exists c E(2_k(t(\vec{x})), b, c) \wedge A(b)))$$

which is, by Definition 1.21,

$$Prog_a A(a) \to A(2_{k+1}(t(\vec{x}))).$$

□

Note how the previous result relies upon successive increases in the complexity of the induction formulae. Thus the restriction in predicative induction is an implicit one in opposition to the usual explicit restrictions in weak theories of arithmetic.

Theorem 1.24. ***A Lower Bound for*** $ProvRec(EA(I;O))$**.**

$$ProvRec(EA(I;O)) \supseteq \mathcal{E}^3.$$

Proof.

We follow the approach in [48]. Let M be an unlimited register machine. We may choose a register machine model which works in unary notation using only the instructions "successor", "predecessor", and "jump". Let f be any elementary function and let P be a program for M which computes f on inputs $\vec{x}$. Since f is an elementary function we may assume P computes f in a number of steps bounded by $2_k(t(\vec{x}))$ for a fixed $k \in \mathbf{N}$ and polynomial t (cf. [14] or [43] for example).

We may choose some scheme to encode the computation by P. Let d be the number of steps performed in the computation and let j be the number of registers used by the program. By convention the computation starts with the inputs $\vec{x} = x_1, \ldots, x_l$ occupying registers $r_1, \ldots, r_j$ (with zeros filling empty registers when $l < j$) and terminates with output b in register r_1. The state of the computation at stage i may be described by the $j+1$-tuple $m(i), r_1(i), \ldots, r_j(i)$ where $m(i)$ is the number of the next machine instruction for stage $i+1$ and $r_1(i), \ldots, r_j(i)$ are the values of the registers $r_1, \ldots, r_j$. Let c be a sequence code of length d such that $(c)_i := \langle m(i), r_1(i), \ldots, r_j(i)\rangle$. Let t be the term $\langle 1, x_1, \ldots, x_l, 0, \ldots, 0\rangle$ for the initial state. We define a bounded formula $C_M(\vec{x}; d, b, c)$ as

$$lh(c) = d + 1 \wedge (c)_0 = t \wedge ((c)_d)_1 = b \wedge \forall i < dA((c)_i, (c)_{i+1}))$$

where $A((c)_i, (c)_{i+1})$ specifies the change in the state of the machine from one stage to the next according to the given program P.

Then $\exists b \exists c C_M(\vec{x}; d, b, c)$ will be progressive in d. The base case is simple to show given the term t for the initial state. For the inductive step, given $(c)_d$, we may choose a term for $(c)_{d+1}$ according to the instruction $m(d)$ in the program P such that $A((c)_d, (c)_{d+1})$ holds and thus $\forall i < d + 1 A((c)_i, (c)_{i+1})$ follows. Where the instruction is "successor" or "predecessor" this is simple to do since those functions are given as terms in our language. Where the instruction is "jump" we may make a case distinction using the cases rule (Lemma 1.8 part 9) and modify $m(d + 1)$ accordingly.

P computes f in a number of steps bounded by $2_k(t(\vec{x}))$ for a fixed $k \in \mathbf{N}$. We may, for this k, choose the following instance of the previous lemma:

$$Prog_d \exists b \exists c C_M(\vec{x}; d, b, c) \rightarrow \exists b \exists c C_M(\vec{x}, 2_k(t(\vec{x})); b, c).$$

As the antecedent is provable we deduce $\exists b \exists c C_M(\vec{x}, 2_k(t(\vec{x})); b, c)$ by a cut. Hence we have an existence condition for f. The uniqueness condition will follow similarly by an induction on d. Therefore f is provably recursive in $EA(I;O)$.

□

Chapter 2

An Infinitary Theory for *EA(I;O)*

2.1 Introduction

We now aim to prove that the provably recursive functions of $EA(I;O)$ are *at most* the elementary functions. This result has appeared before in [18], [46], [47], [48], [68] but only for our particular presentation of $EA(I;O)$ in [66]. (There, unlike the analysis in this chapter, finitary methods are used.) We provide a comprehensive treatment of the result here since the methods we employ provide the groundwork for later chapters. The technique we use is based on a standard proof-theoretic approach where one defines a suitable infinitary calculus into which an arbitrary $EA(I;O)$-proof may be embedded. This new system incorporates an infinitary ω*-rule* for universal quantifications and deals with inductions as potentially transfinite sequences of cuts. As such the system allows for full cut-elimination at the cost of moving to transfinite proof heights. In doing this we are able to provide a uniform measure on the complexity of any $EA(I;O)$-proof and thus calculate the complexity of any provably recursive function in $EA(I;O)$. The process of attaching an ordinal to a theory which in some way measures the strength of the theory dates back to Gentzen's proof of the consistency of Peano Arithmetic [24],[25]. Schütte, [53], later developed methods of analysis using an ω-rule. Our particular approach relates closely to that of Wainer and Fairtlough [19] in their analysis of Peano Arithmetic. In the setting of $EA(I;O)$ we are broadly following the work of Ostrin [46] and also Ostrin and Wainer [47]. It is interesting to note that since inputs in $EA(I;O)$ are never quantified over, $EA(I;O)$ in fact lends itself neatly to a finitary analysis of the type given by Williams in [66]. We shall briefly remark the connections between the two methods in this chapter. Our choice of an infinitary paradigm is driven by the desire to give uniform ordinal bounds on derivations and hence develop a clear comparison between our *slow-growing* theories and the classical *fast-growing* case. Such an approach also serves to simplify matters when we consider the more complex theories in later chapters.

The Infinitary Theory $EA_\infty(I;O)$.

$EA_\infty(I;O)$ is formulated using a Tait-style sequent calculus. We have the same language, term structure and standard definition of numerals as that of $EA(I;O)$ in Chapter 1 except that $EA_\infty(I;O)$ has no free variables. Hence every term t in $EA_\infty(I;O)$ will evaluate to a specific numeral $\bar{k}$ and all formulae are closed. Thus $\Gamma, \Gamma', \ldots$ are now sets of closed formulae. The atomic formulae in $EA_\infty(I;O)$ are just equality and inequality between terms. The logical axioms of $EA_\infty(I;O)$ will be those sets of formulae containing at least one true atom. As such this will incorporate all of the logical and arithmetical axioms on $EA(I;O)$.

We annotate the left of the proof gate with two distinct natural number *declarations* n and m corresponding to an assumption of two separate domains of numbers. A typical logical sequent in $EA_\infty(I;O)$ takes the form

$$n : \mathbb{I}; m : \mathbb{O} \vdash^\alpha \Gamma.$$

This is intended to be read as 'given fixed natural number parameters $\leq n$ from the input domain and given values $\leq m$ from the output domain the truth of Γ (in the standard model) can be established in α-many steps.' For clarity we shall henceforth always use $n; m$ as a shorthand for $n : \mathbb{I}; m : \mathbb{O}$. We use n or n' for input parameters and m, m' or k for output values and separate them using a semi-colon.

We require a way to formally evaluate numerals to use as witnesses for existential quantifiers and must measure the cost involved. For this purpose we include in the calculus *computation* rules. As will be seen the computational fragment of the theory is an independent calculus in its own right since it does not involve any logical formulae or logic cuts. However, the logical fragment of the theory will depend upon incorporating such computations. We shall distinguish computations from logical sequents by using a different notation for the proof gate. A typical computation sequent shall take the form

$$n; m \Vdash^{\alpha} \overline{k}.$$

This should be read as 'given fixed natural number parameters $\leq n$ from the input domain and given values $\leq m$ from the output domain we have a computation of the numeral $\overline{k}$ in α-many steps.' To ease the clutter of notation we shall not from this point forward distinguish numerals using the notation $\overline{k}$ since it should always be clear from the context what is intended: numerals always occur on the right of $\vdash$ and $\Vdash$ proof gates whilst natural number declarations only occur on the left.

Structured Tree-Ordinals

We use *structured tree-ordinals* to measure proof height. Tree-ordinals originate from a constructive or intensional approach to representing countable ordinals dating back to Brouwer and Kleene. The important difference to the usual set-theoretic presentation is that limit ordinals carry additional structure: they are defined by assigning specific *fundamental sequences* to them. Thus any primitive recursive definition of a number-theoretic function can be given over tree-ordinals by extending the definition continuously at limit stages. This is the principal reason for our choice of tree-ordinals. We may add, multiply and exponentiate these ordinals without being tied to normal forms such as Cantor Normal Form. For example our cut-reduction lemma later in the chapter involves simply adding ordinals rather than defining a Hessenberg commutative natural sum.

Tree-ordinals may only be partially ordered since different fundamental sequences may give rise to the same set-theoretic ordinal. For example, if ω is defined by the sequence

$1, 2, 3, \ldots$ and ω' by $0, 1, 2, \ldots$. Therefore we must impose some extra structure on the fundamental sequences which is where the notion of a *structured* tree-ordinal comes into play. A structured tree-ordinal α can be seen as the directed union of

$$\alpha[0] \subseteq \alpha[1] \ldots \subseteq \alpha[n] \subseteq \alpha[n+1] \ldots$$

where for each n, $\alpha[n]$ is a finite subset of α. This will ensure that functions defined over such ordinals really will grow. *Then in our theory, following a rule of inference from proofs of height β we assign the conclusion a height α provided β belongs to $\alpha[n]$ where n is the* input *number parameter in the sequent.*

Here we see why $EA(I;O)$ may lend itself neatly to a finitary analysis. Since inputs are not quantified over, n may be considered as a fixed parameter. This gives rise to a *pointwise* ordinal assignment by the condition $\beta \in \alpha[n]$. The proof heights α, with fixed input parameter n, may be replaced by the finite cardinality of $\alpha[n]$. Hence our system really is *slow-growing* since the Slow-Growing function $G_\alpha(n)$ is equal to the cardinality of $\alpha[n]$ (see Lemma 2.9). However, by using an infinitary measure α we retain uniformity of our results for *all* possible input parameters n. This will also be beneficial for the analysis in the following chapter.

A possible disadvantage associated with tree-ordinals is sensitivity to the choice of fundamental sequence for limits. When collapsing ordinals using the Slow-Growing Hierarchy we may find particular choices of fundamental sequences give rise to much faster or much slower growing functions than we would desire. This problem is exemplified by Weiermann [64]. However we fix the fundamental sequence associated with the limit ordinals we use to be a ‘standard’ one. Indeed we shall use a restricted set of structured tree-ordinals in which we only need to choose a fundamental sequence for ω and we choose the successor function.

The following material comes from [19] to which we direct the reader for a more thorough treatment of the topic including proofs of the facts we quote.

Definition 2.1. *The set Ω of countable* ***tree-ordinals*** *is inductively generated as follows:*

$0 \in \Omega$,

$\alpha \in \Omega \Rightarrow \alpha + 1 := \alpha \cup \{\alpha\} \in \Omega$,

$\forall n \in \mathbf{N}$, $\lambda_n \in \Omega \Rightarrow \lambda := \langle \lambda_n \rangle_{n \in \mathbf{N}} \in \Omega$.

The sub-tree ordering $\prec$ is defined by the transitive closure of $\alpha \prec \alpha + 1$ and for all n, $\lambda_n \prec \lambda$.

We reserve λ to always refer to 'limit ordinals' and use the less formal notation $\lambda = sup_n \lambda_n$ instead of $\lambda = \langle \lambda_n \rangle_{n \in \mathbf{N}}$.

Definition 2.2. *Each ordinal $\alpha \in \Omega$ has, for every $n \in \mathbf{N}$, a finite set $\alpha[n]$ of* ***n-predecessors*** *defined recursively as:*

$0[n] := \varnothing$,

$(\alpha + 1)[n] := \alpha[n] \cup \{\alpha\}$,

$\lambda[n] := \lambda_n[n]$.

Note that our choice of the successor function as the fundamental sequence for ω determines $n \in \omega[n]$ since we have $\omega[n] = (n + 1)[n] = \{n, n - 1, \ldots, 1, 0\}$.

Definition 2.3. *The set Ω^S of* ***structured tree-ordinals*** *contains those $\alpha \in \Omega$ for which*

$$\forall \lambda \preceq \alpha (\forall n \in \mathbf{N} (\lambda_n \in \lambda[n + 1])).$$

Facts 2.4. *Given any $\alpha \in \Omega^S$:*

1. $\beta \prec \alpha \Rightarrow \beta \in \Omega^S$,

2. $\alpha[0] \subseteq \alpha[1] \subseteq \ldots \subseteq \alpha[n] \subseteq \alpha[n + 1] \subseteq \ldots$,

3. $\beta \prec \alpha \Leftrightarrow \beta \in \alpha[n]$ for some $n \in \mathbf{N}$,

4. For $\alpha \neq 0$, the set $\{\beta : \beta \prec \alpha\}$ is well-ordered by $\prec$ and $\beta \prec \alpha \Rightarrow \beta + 1 \preceq \alpha$.

Definition 2.5. ***Addition, multiplication*** *and* ***exponentiation*** *are defined over* Ω *by the following recursions:*

$$\begin{array}{lll} \alpha + 0 := \alpha & \alpha + (\beta + 1) := (\alpha + \beta) + 1 & \alpha + \lambda := sup_n(\alpha + \lambda_n), \\ \alpha \cdot 0 := \alpha & \alpha \cdot (\beta + 1) := (\alpha \cdot \beta) + \alpha & \alpha \cdot \lambda := sup_n(\alpha \cdot \lambda_n), \\ \alpha^0 := 0 + 1 & \alpha^{(\beta+1)} := (\alpha^\beta) \cdot \alpha & \alpha^\lambda := sup_n(\alpha^{\lambda_n}). \end{array}$$

Facts 2.6. *Arithmetic on tree-ordinals preserves structuredness. Given any* α*,* β *and* $\gamma \in \Omega^S$*:*

1. $\gamma \in \beta[n] \Rightarrow \alpha + \gamma \in (\alpha + \beta)[n]$.

2. $\gamma \in \beta[n] \Rightarrow \alpha \cdot \gamma \in (\alpha \cdot \beta)[n]$ *if* $0 \in \alpha[n]$.

3. $\gamma \in \beta[n] \Rightarrow \alpha^\gamma \in (\alpha^\beta)[n]$ *if* $1 \in \alpha[n]$.

4. $\alpha + \beta \in \Omega^S$.

5. $\alpha \cdot \beta \in \Omega^S$ *provided* $0 \in \alpha[1]$.

6. $\alpha^\beta \in \Omega^S$ *provided* $1 \in \alpha[1]$.

Hence, the ordinal functions given above are well-defined on Ω^S.

Definition 2.7. ***The Infinitary Theory*** $EA_\infty(I;O)$.

We are now in a position to define inductively the sequents of $EA_\infty(I;O)$ as follows:

Logical Rules

(L-Ax)

$$n; m \vdash^{\alpha} \Gamma$$

for any α, if Γ contains a true atom.

($\vee$)

$$\frac{n; m \vdash^{\beta} \Gamma, A_i}{n; m \vdash^{\alpha} \Gamma, A_0 \vee A_1}$$

if $\beta \in \alpha[n]$ and where $i = 0$ or 1.

($\wedge$)

$$\frac{n; m \vdash^{\beta_0} \Gamma, A_0 \qquad n; m \vdash^{\beta_1} \Gamma, A_1}{n; m \vdash^{\alpha} \Gamma, A_0 \wedge A_1}$$

if $\beta_0, \beta_1 \in \alpha[n]$.

($\forall$)

$$\frac{\{n; \mathrm{max}(m, k) \vdash^{\beta_k} \Gamma, A(k)\}_{k \in \mathbb{N}}}{n; m \vdash^{\alpha} \Gamma, \forall a A(a)}$$

if for all $k \in \mathbf{N}$ $\beta_k \in \alpha[n]$.

($\exists$)

$$\frac{n; m \Vdash^{\beta_0} k \qquad n; m \vdash^{\beta_1} \Gamma, A(k)}{n; m \vdash^{\alpha} \Gamma, \exists a A(a)}$$

if $\beta_0 \in \beta_1[n]$ and $\beta_1 \in \alpha[n]$.

(L-Cut)

$$\frac{n; m \vdash^{\beta_0} \Gamma, \neg C \qquad n; m \vdash^{\beta_1} \Gamma, C}{n; m \vdash^{\alpha} \Gamma}$$

if $\beta_0, \beta_1 \in \alpha[n]$.

(C-Cut)

$$\frac{n; m \Vdash^{\beta_0} k \qquad n; k \vdash^{\beta_1} \Gamma}{n; m \vdash^{\alpha} \Gamma}$$

if $\beta_0, \beta_1 \in \alpha[n]$.

Computational Rules

($\mathbb{O}$-Ax)

$$n; m \Vdash^{\alpha} k$$

for any α, if $k \leq p(m)$.

($\mathbb{O}$-Cut)

$$\frac{n; m \Vdash^{\beta_0} m' \qquad n; m' \Vdash^{\beta_1} k}{n; m \Vdash^{\alpha} k}$$

if $\beta_0, \beta_1 \in \alpha[n]$.

Remarks

The $(\exists)$ and (C-Cut) rules are where $\Vdash$ computations influence $\vdash$ derivations. We call the (C-Cut) rule a *computational cut* whilst the (L-Cut) rule is a *logical cut*. The polynomial in the computational axiom ($\mathbb{O}$-Ax) is chosen to bound the values of the term constructors of $EA(I;O)$ on the output value m. As such it is easy to see that the quadratic $2(m+1)^2$ will suffice. We note that in the existential rule we have made a stronger requirement on the tree-ordinal conditions in an approach taken from Williams [66]. By insisting $\beta_0 \in \beta_1[n]$ as well as $\beta_1 \in \alpha[n]$ we put in place additional structure which simplifies the cut-reduction lemma later in the chapter. This has no significant effect on the length of derivations.

At this point we may note the similarities between the system we are presenting and other approaches we cited at the start of the chapter. $EA_\infty(I;O)$ bears a close resemblance to the analysis given by Ostrin [46] and Ostrin and Wainer [47]. There are only minor differences such as the separation of the computation and logic rules which seeks to provide clarity. We may also make comparisons with the calculus of Fairtlough and Wainer [19] p.166 in their analysis of Peano Arithmetic and its fragments. Roughly speaking if we were to drop the distinction between input and output parameters we are left with their system. However we note that this leads to a *fast-growing* system since the parameter n changes in their universal quantification rule rather than staying fixed in ours. Our ω-rule requires $\beta \in \alpha[n]$ for a fixed n rather than $\beta \in \alpha[\max(n,k)]$. This is crucial in restricting the strength of the theory.

The Slow-Growing Nature of $EA_\infty(I;O)$

We briefly return to a comment made earlier that the system $EA_\infty(I;O)$ really is a *slow-growing* infinitary theory.

Definition 2.8. *For $\alpha \in \Omega^S$ the* ***Slow-Growing Hierarchy*** *of functions $G_\alpha : \mathbf{N} \to \mathbf{N}$ are defined by* $G_\alpha(n) := \text{card } \alpha[n]$ *and hence*

$$\begin{array}{rcl} G_0(n) & = & 0, \\ G_{\alpha+1}(n) & = & G_\alpha(n) + 1, \\ G_\lambda(n) & = & G_{\lambda_n}(n). \end{array}$$

For any $\alpha \in \Omega^S$ note that $G_\alpha(n)$ is simply the function result from replacing ω by $n+1$ throughout α. This follows from our choice of the successor function as the fundamental sequence for ω. The following lemma justifies the claim that proof heights α, with fixed input parameter n, may be replaced by the finite cardinality of $\alpha[n]$.

Lemma 2.9.

$$n; m \vdash^\alpha \Gamma \qquad \Rightarrow \qquad n; m \vdash^{G_\alpha(n)} \Gamma.$$

Proof.

We use induction over the derivation of Γ with a case distinction according to the final rule of inference used. If $n; m \vdash^\alpha \Gamma$ is an instance of (L-Ax) then the tree-ordinal α is arbitrary. Hence $n; m \vdash^{G_\alpha(n)} \Gamma$ is also an instance of (L-Ax). Otherwise we have some rule of inference from premise(s) of the form $n; m' \vdash^{\beta_i} \Gamma'$ where $\beta_i \in \alpha[n]$ for each i. Applying the induction hypothesis yields $n; m' \vdash^{G_{\beta_i}(n)} \Gamma'$. Then as $\beta_i \in \alpha[n]$ implies $\beta_i[n] \subset \alpha[n]$ we find $G_{\beta_i}(n) < G_\alpha(n)$ and thus $G_{\beta_i}(n) \in (G_\alpha(n))[n]$. Hence we may re-apply the same rule concluding $n; m \vdash^{G_\alpha(n)} \Gamma$. Note that where a computational sequent is present in the premises of the rule we shall need a sub-induction to show a computation of height α entails one of height $G_\alpha(n)$. The argument is exactly the same.

$\square$

Note that the converse is not necessarily true. For example $n+1 \in (G_{\omega+1}(n))[n]$ but $n+1 \notin \omega + 1[n]$. Thus using a tree-ordinal measure of height produces uniformity in n and this will be crucial for analyses in later chapters.

2.2 Computations in $EA_\infty(I;O)$

We look to find a suitable bounding function for the computation rules in $EA_\infty(I;O)$. Then where an existential rule has been used we may estimate an upper bound on the value of the witnessing term.

Definition 2.10. *The set of* ***exponential tree-ordinals*** *denoted by $E(\omega)$ are generated inductively by*

$$\begin{array}{rcl} 0 & \in & E(\omega), \\ \omega & \in & E(\omega), \\ \alpha, \beta \in E(\omega) & \Rightarrow & \alpha + \beta \in E(\omega), \\ \alpha \in E(\omega) & \Rightarrow & 2^\alpha \in E(\omega). \end{array}$$

All tree-ordinals in the set $E(\omega)$ are structured by Facts 2.6 and $E(\omega)$ is clearly closed under successors and multiplication by a finite tree-ordinal. $G_\alpha(n)$ is the function obtained from replacing ω by $n+1$ throughout α. Hence, if $\alpha \in E(\omega)$ then $G_\alpha(n)$ is an exponential polynomial in n and therefore an elementary function in n.

Definition 2.11. *For $\alpha \in \Omega^S$ the functions $B_\alpha : \mathbf{N}^2 \to \mathbf{N}$ are defined using the following recursion scheme:*

$$\begin{array}{rcl} B_0(n;m) & := & p(m), \\ B_{\alpha+1}(n;m) & := & B_\alpha(n; B_\alpha(n;m)), \\ B_\lambda(n;m) & := & B_{\lambda_n}(n;m). \end{array}$$

where $p(m) := 2(m+1)^2$ is the same polynomial as used in the ($\mathbb{O}$-Ax) rule.

Note in the definition of B_α that substitution at successor stages occurs *after* the semi-colon whilst diagonalisation uses the argument *before* the semi-colon. This is clearly in keeping with the spirit of Bellantoni-Cook recursion schemes.

Lemma 2.12. *Define the iterates of $p(m)$ as $p^0(m) = m$ and $p^{k+1}(m) = p(p^k(m))$. Then for $\alpha \in \Omega^S$ we have*

$$B_\alpha(n;m) = p^k(m) \quad \textit{where } k := 2^{G_\alpha(n)}.$$

Hence if $\alpha \in E(\omega)$ *then* $B_\alpha(n; m)$ *is an elementary function in* $n; m$.

Proof.

The first claim follows by induction on α. The zero case and limit cases are trivial by the definitions of $G_\alpha(n)$ and $B_\alpha(n; m)$. When α is a successor the induction hypothesis implies

$$B_{\alpha+1}(n; m) = B_\alpha(n; B_\alpha(n; m)) = p^k(p^k(m)) = p^{k+k}(m)$$

where $k := 2^{G_\alpha(n)}$. As $2^{G_\alpha(n)} + 2^{G_\alpha(n)} = 2^{G_{\alpha+1}(n)}$ the result follows.

For the second claim we know $G_\alpha(n)$ is an exponential polynomial in n for $\alpha \in E(\omega)$. The function $p(m)$ is sub-elementary thus the iterate $p^k(m)$ is an elementary function in k and m (cf. [51]). Then $B_\alpha(n; m)$, for $\alpha \in E(\omega)$, may be defined by composition of two elementary functions and hence is an elementary function in n and m.

□

We shall require these functions to have certain majorization properties. Given a function $f(n)$ we say f is positive if $n \leq f(n)$ and strictly positive if $n < f(n)$. For $n < n'$ we say f is increasing if $f(n) \leq f(n')$ and strictly increasing if $f(n) < f(n')$.

Lemma 2.13. *For* $\alpha, \beta \in \Omega^S$ *and* $m, n \in \mathbf{N}$*:*

1. *If* $\beta \in \alpha[n]$ *then* $G_\beta(n) < G_{\beta+1}(n) \leq G_\alpha(n)$.
2. $G_\alpha(n)$ *is increasing in* n*, strictly so if* α *is infinite.*
3. $G_\alpha(n) \leq G_{\beta+\alpha}(n)$.
4. *If* $\beta \in \alpha[n]$ *then* $B_\beta(n; m) < B_{\beta+1}(n; m) \leq B_\alpha(n; m)$.
5. $B_\alpha(n; m)$ *is strictly increasing in* m *and increasing in* n*, strictly so if* α *is infinite.*
6. $B_\alpha(n; m) \leq B_{\beta+\alpha}(n; m)$.

Proof.

Recall that $G_\alpha(n) := \text{card } \alpha[n]$. Since $\beta \in \alpha[n]$ implies $\beta[n] \subset \beta + 1[n] \subseteq \alpha[n]$

part 1 follows. When $n < n'$ we have $\alpha[n] \subset \alpha[n']$ if α is infinite and $\alpha[n] \subseteq \alpha[n']$ otherwise. This gives part 2. For part 3 note that as addition extends continuously at limits: $G_{\beta+\alpha}(n) = G_\beta(n) + G_\alpha(n)$.

Recall $p(m) := 2(m+1)^2$. Then since $p(m)$ is both strictly increasing and strictly positive the function $p^k(m)$ is strictly increasing in m and k. Parts 4, 5 and 6 will then follow as $B_\alpha(n;m) = p^k(m)$ with $k := 2^{G_\alpha(n)}$.

□

Lemma 2.14. ***Bounding for*** $EA_\infty(I;O)$**.**

$$n;m \Vdash^\alpha k \qquad \Leftrightarrow \qquad k \leq B_\alpha(n;m).$$

Proof.

Tackling the left to right implication first we use induction over the derivation of the computational sequent.

If the derivation is an instance of ($\mathbb{O}$-Ax) then $k \leq p(m) = B_0(n;m)$. If α is zero we are done. Otherwise $0 \in \alpha[n]$ and we apply Lemma 2.13 part 4 which shows $B_0(n;m) \leq B_\alpha(n;m)$.

The only other possibility is that the derivation results from ($\mathbb{O}$-Cut) such as

$$\frac{n;m \Vdash^{\beta_0} m' \qquad n;m' \Vdash^{\beta_1} k}{n;m \Vdash^\alpha k}$$

where $\beta_0, \beta_1 \in \alpha[n]$. By the structure imposed on the tree-ordinals we know from $\beta_0, \beta_1 \in \alpha[n]$ that either $\beta_0 \in \beta_1[n]$ or $\beta_1 \in \beta_0[n]$ or $\beta_0 = \beta_1$. Letting $\beta := \max(\beta_0, \beta_1)$ the induction hypothesis for α and Lemma 2.13 part 4 give

$$k \leq B_{\beta_1}(n;m') \leq B_\beta(n;m') \qquad \text{and} \qquad m' \leq B_{\beta_0}(n;m) \leq B_\beta(n;m).$$

Now appealing to Lemma 2.13 parts 4 and 5 and the definition of B_α we conclude

$$k \leq B_\beta(n;m') \leq B_\beta(n;B_\beta(n;m)) = B_{\beta+1}(n;m) \leq B_\alpha(n;m).$$

We may now turn to the right to left implication using induction over α. Assume that $\alpha = 0$. Then $k \leq B_0(n;m) = p(m)$. The result follows immediately by ($\mathbb{O}$-Ax).

Now assume that α is the successor $\beta + 1$ so that $k \leq B_{\beta+1}(n;m) = B_\beta(n; B_\beta(n;m))$. Letting $m' := B_\beta(n;m)$ the induction hypothesis for α gives us two derivations

$$n;m \Vdash^\beta m' \qquad \text{and} \qquad n;m' \Vdash^\beta k.$$

Since $\beta \in \alpha[n]$ the result follows by applying ($\mathbb{O}$-Cut).

Finally assume that α is a limit $sup_n(\lambda_n)$ so that $k \leq B_\lambda(n;m) = B_{\lambda_n}(n;m)$. Then for every n we may apply the induction hypothesis to give

$$n;m \Vdash^{\lambda_n} k$$

and the result follows using a simple sub-induction to change the proof height from λ_n to λ. The axiom case is self-evident. Now assume an application of ($\mathbb{O}$-Cut) gives proof height λ_n from premises of heights β_i. Then since $\beta_i \in \lambda_n[n] = \lambda[n]$ we may have in each case taken the new height to be λ. Hence

$$n;m \Vdash^\lambda k.$$

□

Corollary 2.15. ***Weakening for Computations.*** *If we have the computation*

$$n;m \Vdash^\alpha k$$

and if $n \leq n', m \leq m', k' \leq k$ and $\alpha[n] \subseteq \alpha'[n]$ then for any $\gamma \in \Omega^S$ we also have the computation

$$n';m' \Vdash^{\gamma+\alpha'} k'.$$

Proof.

From the bounding result above we know $k \leq B_\alpha(n;m)$. If $\alpha[n] \subseteq \alpha'[n]$ then

$G_\alpha(n) \leq G_{\alpha'}(n)$. Hence, given the assumptions, applying Lemma 2.12 and Lemma 2.13 we find $k' \leq k \leq B_\alpha(n;m) \leq B_{\gamma+\alpha'}(n';m')$. Applying the bounding result again gives $n';m' \Vdash^{\gamma+\alpha'} k'$.

□

We shall now look at how to compute specific values of terms from $EA(I;O)$. We define the notion of the height $|t|$ of a term t as the natural number corresponding to how many applications of the term constructors are applied to base terms 0, x_j or a_j.

Definition 2.16. *We define the* ***height of term*** *t in $EA(I;O)$, denoted $|t|$ inductively as*

$$\begin{array}{lll} |t| & := & 0 \qquad\qquad\qquad \textit{if } t = 0,\ t = a \textit{ or } t = x. \\ |t| & := & \max(|t_i|) + 1 \quad \textit{if } t \textit{ results from applying a function symbol to sub-terms } t_i. \end{array}$$

Lemma 2.17. *For any term $t(\vec{x};\vec{a})$ in $EA(I;O)$ and any numbers m and n, if the value of the term upon substituting $m_i \leq m$ for each a_i and $n_i \leq n$ for each x_i is the number k we have in $EA_\infty(I;O)$ the computation*

$$n;m \Vdash^{\omega\cdot(|t|+1)} k.$$

Proof.

We use induction on $|t|$ to show that $k \leq B_{\omega\cdot(|t|+1)}(n;m)$ whence the result follows by applying the bounding lemma (2.14). Recall that $p(m)$ bounds the values of the term constructors and is strictly positive. Hence, if $|t| = 0$ then

$$k \leq \max(n;m) \leq p^{2^{n+1}}(m) = B_\omega(n;m).$$

Now assume $|t| > 0$. Then the result holds for $\max(|t_i|)$ where t_i are the immediate sub-terms of t. Let $m' := B_{\omega\cdot(\max(|t_i|)+1)}(n;m) = B_{\omega\cdot|t|}(n;m)$. Then as the values of the sub-terms are less than or equal to m' we find $k \leq p(m')$. Applying Lemma 2.13

$$p(m') \leq B_{\omega\cdot|t|}(n;m') = B_{\omega\cdot|t|}(n;B_{\omega\cdot|t|}(n;m)) = B_{\omega\cdot|t|+1}(n;m) \leq B_{\omega\cdot(|t|+1)}(n;m).$$

□

2.3 Structural Rules for $EA_\infty(I;O)$

In this section we provide structural rules for weakening, conjunction inversion and universal inversion which are required for the cut-elimination and embedding processes later.

Lemma 2.18. ***Weakening for Logical Rules.*** *If we have a derivation*

$$n;m \vdash^{\alpha} \Gamma$$

and if $n \leq n', m \leq m', \Gamma \subseteq \Gamma'$ *and* $\alpha[n] \subseteq \alpha'[n]$ *then for any* $\gamma \in \Omega^S$ *we also have the derivation*

$$n';m' \vdash^{\gamma+\alpha'} \Gamma'.$$

Proof.

We shall use induction over the derivation of Γ with a case distinction according to which rule is applied last. We shall only need to look in detail at particular cases since the others use exactly the same reasoning. Throughout we make use of the fact that $\beta \in \alpha[n] \Rightarrow \gamma + \beta \in (\gamma + \alpha)[n]$ for any β, α and $\gamma \in \Omega^S$, cf. 2.6.

1. (L-Ax). If Γ contains a true atom then any Γ' for which $\Gamma \subseteq \Gamma'$ also contains that atom. The tree-ordinal height in the (L-Ax) rule is arbitrary so may be taken to be $\gamma + \alpha'$. Likewise the numerical values in the declarations may be increased and the atom in question is still true. Hence $n';m' \vdash^{\gamma+\alpha'} \Gamma'$ is also an instance of the (L-Ax).

2. $(\vee)$, $(\wedge)$, $(\forall)$ and (L-Cut). These four cases are almost identical. For each we seek to apply the induction hypothesis to the premise(s) and then re-apply the relevant rule. For example, in the case of $(\vee)$ let us assume that $\Gamma :\equiv \Gamma_0, A_0 \vee A_1$. After applying the induction hypothesis to the premise of the rule we have the derivation

$$n';m' \vdash^{\gamma+\beta} \Gamma', A_i$$

for $i = 0$ or 1 and some $\beta \in \alpha[n]$. Hence as $\beta \in \alpha[n] \subseteq \alpha'[n] \subseteq \alpha'[n']$ we may conclude that $\gamma + \beta \in (\gamma + \alpha')[n']$. We re-apply the rule deriving

$$n'; m' \vdash^{\gamma+\alpha'} \Gamma', A_0 \vee A_1.$$

Since $A_0 \vee A_1 \subseteq \Gamma \subseteq \Gamma'$ this is equivalent to

$$n'; m' \vdash^{\gamma+\alpha'} \Gamma'.$$

3. $(\exists)$ and (C-Cut). We follow the same reasoning as above with two minor differences. Firstly both these rules have a computation as the left-hand premise. Hence we only apply the induction hypothesis to the right-hand premises. To deal with the left-hand premises we use Corollary 2.15.

In the case of (C-Cut) the required conditions on the proof heights to re-apply the rule follow the reasoning of the previous case. From the original premises of the rule we have

$$\cfrac{\begin{matrix} n; m \Vdash^{\beta_0} k & \qquad n; k \vdash^{\beta_1} \Gamma \\ \text{[Corollary 2.15]} & \text{[IH]} \\ n'; m' \Vdash^{\gamma+\beta_0} k & \qquad n'; k \vdash^{\gamma+\beta_1} \Gamma' \end{matrix}}{n'; m' \vdash^{\gamma+\alpha'} \Gamma'} \text{ (C-Cut)}$$

The second difference comes in the case of the existential rule where there is extra structure on the conditions governing the proof heights. That is we have $\beta_0 \in \beta_1[n]$ and $\beta_1 \in \alpha[n]$. Since $n \leq n'$ we find $\beta_1[n] \subseteq \beta_1[n']$ and hence $\gamma + \beta_0 \in (\gamma + \beta_1)[n']$. We also find, as before, that $\beta_1 \in \alpha[n] \subseteq \alpha'[n]$ implies $\gamma + \beta_1 \in (\gamma + \alpha')[n']$. Hence letting $\Gamma :\equiv \Gamma_0, \exists a A(a)$ we find

$$\cfrac{\begin{matrix} n; m \Vdash^{\beta_0} k & \qquad n; m \vdash^{\beta_1} \Gamma_0, A(k) \\ \text{[Corollary 2.15]} & \text{[IH]} \\ n'; m' \Vdash^{\gamma+\beta_0} k & \qquad n'; m' \vdash^{\gamma+\beta_1} \Gamma', A(k) \end{matrix}}{n'; m' \vdash^{\gamma+\alpha'} \Gamma'} \text{ } (\exists)$$

□

Lemma 2.19. ***Conjunction inversion.***

$$n;m \vdash^{\alpha} \Gamma, A_0 \wedge A_1 \quad \Rightarrow \quad n;m \vdash^{\alpha} \Gamma, A_i \qquad \textit{where } i = 0 \textit{ or } 1.$$

Proof.

Again we use induction over the derivation of $\Gamma, A_0 \wedge A_1$. There are two main cases where either $A_0 \wedge A_1$ is the principal formula in the last rule applied or where the last rule applied is on Γ with $A_0 \wedge A_1$ a side formula.

1. Firstly note that if the derivation is an axiom then Γ contains a true atomic formula. Hence $n;m \vdash^{\alpha} \Gamma, A_i$ is also an axiom where $i = 0$ or 1.

2. Assume the last rule in the derivation has principal formula in Γ with $A_0 \wedge A_1$ a side formula. Inductively from the premise(s) we have a derivation or derivations of the form

$$n;m' \vdash^{\beta_k} \Gamma', A_i$$

where $i = 0$ or 1 and $\beta_k \in \alpha[n]$. From here we may simply re-apply the rule replacing Γ' by Γ.

3. The only other possibility is that the last rule applied is ($\wedge$) in which $A_0 \wedge A_1$ is the principal formula. Then we would have the premises

$$n;m \vdash^{\beta_0} \Gamma', A_0 \qquad n;m \vdash^{\beta_1} \Gamma', A_1$$

where $\beta_0, \beta_1 \in \alpha[n]$. In these derivations Γ' may contain $A_0 \wedge A_1$ itself since contraction may occur after the rule is applied. Hence we must apply the induction hypothesis to the relevant premise in order to invert $A_0 \wedge A_1$ and then contract any addition instances of A_i. Then since $\beta_i[n] \subseteq \alpha[n]$ we have the required result by weakening 2.18. If Γ' does not contain $A_0 \wedge A_1$ then simply applying weakening to the tree-ordinal bound on the relevant premise gives the result.

□

Lemma 2.20. ***Universal quantifier inversion.***

$$n;m \vdash^{\alpha} \Gamma, \forall a A(a) \quad \Rightarrow \quad n;\max(m,k) \vdash^{\alpha} \Gamma, A(k) \quad \textit{for every } k \in \mathbf{N}.$$

Proof.

We follow the same approach as 2.19. In the axiom case or in the case of any rule of inference with principal formula within Γ the argument is essentially the same. The only minor difference occurs in the case of a computational cut. From the premises, with $\beta_0, \beta_1 \in \alpha[n]$, we have

$$\dfrac{\dfrac{n;m \Vdash^{\beta_0} k'}{n;\max(m,k) \Vdash^{\beta_0} k'}\,[\text{Corollary 2.15}] \qquad \dfrac{n;k' \vdash^{\beta_1} \Gamma, \forall a A(a)}{n;\max(m,k,k') \vdash^{\beta_1} \Gamma, A(k)}\,[\text{IH}]}{n;\max(m,k) \vdash^{\alpha} \Gamma, A(k)}\ \text{(C-Cut)}$$

Where the last rule applied is ($\forall$) and $\forall a A(a)$ is principal we have a premise for each $k \in \mathbf{N}$ of the form

$$n;\max(m,k) \vdash^{\beta_k} \Gamma', A(k)$$

where $\beta_k \in \alpha[n]$. Two sub-cases occur dependant upon whether or not Γ' contains the principal formula $\forall a A(a)$. In the first case we must apply the induction hypothesis in order to replace Γ' by Γ and then weaken the tree-ordinal bound by Lemma 2.18 since $\beta_k \in \alpha[n]$. In the second case we may simply use weakening to change the tree-ordinal height to α.

□

2.4 Cut-Elimination for $EA_{\infty}(I;O)$

We now show how to eliminate logical cuts from a proof in $EA_{\infty}(I;O)$ at the cost of successive exponential increases in the proof height. The methods used in this section are standard and we refer the reader to either Fairtlough and Wainer [19] or Ostrin [46] in the setting of a two sorted theory.

Definition 2.21. *The* ***height of a formula*** A*, denoted* $|A|$*, is defined inductively as:*

$$\begin{array}{rcll} |A| & := & 1 & \textit{if } A \textit{ is atomic.} \\ |A_0 \Diamond A_1| & := & \mathit{max}(|A_0|, |A_1|) + 1 & \textit{if } \Diamond \textit{ is } \vee \textit{ or } \wedge. \\ |\Diamond A(a)| & := & |A| + 1 & \textit{if } \Diamond \textit{ is } \exists a \textit{ or } \forall a. \end{array}$$

Definition 2.22. *The* ***cut-rank*** r *of a derivation in* $EA_\infty(I;O)$ *is defined to be the supremum of all the heights* $|C|$ *of the cut-formulas* C *occurring in the derivation in question.*

The cut-rank r is noted as a subscript to the right of the proof gate thus: $\vdash^\alpha_r$. Note that a cut-free proof in $EA_\infty(I;O)$ denoted by $\vdash^\alpha_0$ only means that there are no (L-Cuts) present. We may still have many computational (C-Cut) rules present. The computational proof gate need not mention the cut-rank since there are no (L-Cuts) involved in $\Vdash$ derivations. Also note that the structural rules given in the previous section do not have any effect on the cut-rank of derivations. The same is true of the following lemma.

Lemma 2.23. ***False Atom.*** *If* C *is atomic and true so that* $\neg C$ *is false then*

$$n;m \vdash^\alpha \Gamma, \neg C \quad \Rightarrow \quad n;m \vdash^\alpha \Gamma.$$

Proof.

We use induction over the derivation of $\Gamma, \neg C$. Note that no rule, other than cut, may conclude $\neg C$ since C is atomic. Where the principal formula of the last rule applied is contained within Γ or where the rule is (L-Cut) we may simply apply the induction hypothesis to the premise(s) and re-apply the same rule. The only other possibility is that $\Gamma, \neg C$ itself is an instance of (L-Ax). Then the axiom must occur inside Γ because $\neg C$ is false. Hence the result follows from another instance of (L-Ax).

□

Lemma 2.24. ***Cut-Rank Reduction.*** *Assume that in $EA_\infty(I;O)$ we have the derivations*

$$n;m \vdash^{\alpha}_{r} \Gamma_0, C \quad \textit{and} \quad n;m \vdash^{\beta}_{r} \Gamma_1, \neg C$$

where C is either an atom or of the form $C_0 \vee C_1$ or $\exists a C_0(a)$ with $|C|$=r+1. Further assume that $\alpha[n] \subseteq \beta[n]$. Then

$$n;m \vdash^{\beta+\alpha}_{r} \Gamma_0, \Gamma_1.$$

Proof.

We proceed by induction over the derivation of Γ_0, C in the first assumption.

1. Consider the case where C is a side formula of the last rule applied so that the principal formula resides within Γ_0.

i. Assume that

$$n;m \vdash^{\alpha}_{r} \Gamma_0, C$$

is an instance of (L-Ax). Then since C is a only a side formula the required result is also an instance of (L-Ax).

ii. Otherwise we have a rule of inference with premise(s) of the form

$$n;m' \vdash^{\gamma}_{r} \Gamma'_0, C \tag{2.1}$$

where $\gamma \in \alpha[n]$. Such a derivation may have involved a computational sequent as a left-hand premise. Furthermore, m' may be greater than or equal to m if the rule is (∀) or (C-Cut) or we may find that $m' < m$ in the case of (C-Cut).

Assume $m' \geq m$. Then applying weakening to the second assumption of the lemma to increase the output declaration

$$n;m' \vdash^{\beta}_{r} \Gamma_1, \neg C. \tag{2.2}$$

Having assumed that $\alpha[n] \subseteq \beta[n]$ we have $\gamma[n] \subseteq \beta[n]$ and may apply the induction hypothesis to (2.1) and (2.2) giving

$$n;m' \vdash^{\beta+\gamma}_{r} \Gamma'_0, \Gamma_1.$$

Then since $\beta+\gamma \in \beta+\alpha[n]$ we re-apply the rule (using the computational sequent again if necessary) giving

$$n; m \vdash_r^{\beta+\alpha} \Gamma_0, \Gamma_1.$$

If it is the case that $m' < m$ then the rule must have been (a vacant application of) a computational cut and $\Gamma_0' \equiv \Gamma_0$. We apply weakening first to change the output declaration in (2.1) to m. Then by the induction hypothesis with the second assumption of the lemma we have

$$n; m \vdash_r^{\beta+\gamma} \Gamma_0, \Gamma_1.$$

All that remains is to apply weakening to change the proof height from $\beta + \gamma$ to $\beta + \alpha$.

2. Now we consider the cases where C is the principal formula of the last rule applied. We only have three sub-cases where C is atomic or of the form $C_0 \vee C_1$ or $\exists a C_0(a)$.

i. Where C is atomic and principal, $\neg C$ is false so we may appeal to Lemma 2.23 above for the second assumption which shows

$$n; m \vdash_r^{\beta} \Gamma_1, \neg C \quad \Rightarrow \quad n; m \vdash_r^{\beta} \Gamma_1.$$

The result now follows by weakening: changing the height to $\beta + \alpha$ and adding the formulae Γ_0.

ii. Now assume the last rule applied is ($\vee$) so that we have the following premise to the first assumption with $\gamma \in \alpha[n]$:

$$n; m \vdash_r^{\gamma} \Gamma_0, C_0 \vee C_1, C_i \qquad \text{for } i = 0 \text{ or } 1.$$

Here, like the following case, we explicitly include the possibility that the principal formula $C_0 \vee C_1$ occurs in the premise or else add the principal formula by weakening. At this point, since $\gamma \in \alpha[n]$ ensures $\gamma[n] \subseteq \beta[n]$, we apply the induction hypothesis with the second assumption of the lemma. This gives

$$n; m \vdash_r^{\beta+\gamma} \Gamma_0, \Gamma_1, C_i \qquad \text{for } i = 0 \text{ or } 1. \tag{2.3}$$

Looking again at the second assumption of the lemma we have

$$n;m \vdash^{\beta}_{r} \Gamma_1, \neg C_0 \wedge \neg C_1.$$

Now using Lemma 2.19 to invert the conjunction and adding Γ_0 through weakening we obtain

$$n;m \vdash^{\beta}_{r} \Gamma_0, \Gamma_1, \neg C_i \qquad \text{for } i = 0 \text{ or } 1. \tag{2.4}$$

Since $\gamma \in \alpha[n]$, we have both $\beta + \gamma \in (\beta + \alpha)[n]$ and $\beta \in \beta + \alpha[n]$. Hence by (L-Cut) on 2.3 and 2.4, knowing that the cut-rank will remain r since $|C_i| \leq r$,

$$n;m \vdash^{\beta+\alpha}_{r} \Gamma_0, \Gamma_1.$$

iii. If $C :\equiv \exists a C_0(a)$ is principal in an $(\exists)$ rule, then the premises to the first assumption of the lemma are

$$n;m \Vdash^{\gamma_0} k \qquad n;m \vdash^{\gamma_1}_{r} \Gamma_0, \exists a C_0(a), C_0(k)$$

with $\gamma_0 \in \gamma_1[n]$ and $\gamma_1 \in \alpha[n]$. We apply the induction hypothesis to the right-hand premise of the rule together with the second assumption of the lemma to find

$$n;m \vdash^{\beta+\gamma_1}_{r} \Gamma_0, \Gamma_1, C_0(k). \tag{2.5}$$

The second assumption reads

$$n;m \vdash^{\beta}_{r} \Gamma_1, \forall a \neg C_0(a).$$

Using weakening and universal inversion we obtain, for all $k \in \mathbf{N}$,

$$n; \max(m,k) \vdash^{\beta}_{r} \Gamma_0, \Gamma_1, \neg C_0(k). \tag{2.6}$$

Since we have the computation $n;m \Vdash^{\gamma_0} k$ as a premise to the $(\exists)$ rule and by 2.15 $n;m \Vdash^{\gamma_0} m$, we have by our bounding result

$$n;m \Vdash^{\beta+\gamma_0} \max(m,k). \tag{2.7}$$

Recalling that $\gamma_0 \in \gamma_1[n]$ from the finer conditions on the existential rule, a (C-Cut) on 2.6 and 2.7 gives

$$n; m \vdash_r^{\beta+\gamma_1} \Gamma_0, \Gamma_1, \neg C_0(k) \tag{2.8}$$

since $\beta, \beta + \gamma_0 \in \beta + \gamma_1[n]$. Finally using $\beta + \gamma_1 \in \beta + \alpha[n]$ we apply (L-Cut) on 2.5 and 2.8 with cut-formula $C_0(k)$ of height r:

$$n; m \vdash_r^{\beta+\alpha} \Gamma_0, \Gamma_1.$$

□

Theorem 2.25. ***Cut-Elimination.***

$$n; m \vdash_{r+1}^{\alpha} \Gamma \quad \Rightarrow \quad n; m \vdash_r^{2^\alpha} \Gamma.$$

Proof.

Again by induction over the derivation of $n; m \vdash_{r+1}^{\alpha} \Gamma$.

1. If Γ is an axiom then the result follows trivially since the ordinal bound and cut-rank are arbitrary in the axiom rule.

2. If Γ comes about via a rule of inference which is not a cut of rank $r + 1$ then assume the premise(s) have height(s) β_i. These derivations may have cuts of rank $r+1$. We apply the induction hypothesis giving proof height(s) 2^{β_i} and reducing the rank by 1. Now since $\beta_i \in \alpha[n]$ we have $2^{\beta_i} \in 2^\alpha[n]$ by 2.6 part 3. Hence the same rule may be re-applied to give the result.

3. If Γ comes about via a cut on C with $|C| = r + 1$ we have for $\beta_0, \beta_1 \in \alpha[n]$

$$n; m \vdash_{r+1}^{\beta_0} \Gamma, C \qquad n; m \vdash_{r+1}^{\beta_1} \Gamma, \neg C.$$

Letting $\beta := \max(\beta_0, \beta_1) \in \alpha[n]$ and weakening one of the premises accordingly we apply the induction hypothesis to give

$$n; m \vdash_r^{2^\beta} \Gamma, C \qquad n; m \vdash_r^{2^\beta} \Gamma, \neg C.$$

Now the cut-reduction lemma (2.24) applies because by symmetry one of C or $\neg C$ must be of the required form. This leaves

$$n;m \vdash_r^{2^\beta+2^\beta} \Gamma.$$

Now either $2^\beta + 2^\beta = 2^\alpha$ else we may apply weakening to change the height to 2^α since $(2^\beta + 2^\beta)[n] \subseteq 2^\alpha[n]$.

$\square$

Corollary 2.26. ***Full Cut-Elimination.*** *If we define $2_0(\alpha) := \alpha$ and $2_{r+1}(\alpha) := 2^{2_r(\alpha)}$ then letting $\delta := 2_r(\alpha)$ we have*

$$n;m \vdash_r^\alpha \Gamma \quad \Rightarrow \quad n;m \vdash_0^\delta \Gamma.$$

Proof.

We use induction over r. If $r = 0$ there is nothing to do. For the successor case assume $r = r' + 1$. We apply the cut-elimination theorem above to reduce the rank by 1 and then apply the induction hypothesis. Since $2_{r'}(2^\alpha) = 2_{r'+1}(\alpha)$ we attain the required result. Hence we may fully eliminate logical cuts of maximum rank r from a given derivation in $EA_\infty(I;O)$ at the cost of a r-times iterated exponential increase in the proof height.

$\square$

2.5 Embedding of $EA(I;O)$

In this section we show that any $EA(I;O)$ derivation may be embedded into $EA_\infty(I;O)$ at the cost of moving to a transfinite height where inductions are replaced by sequences of cuts. Henceforth, given an $EA(I;O)$ formula $A(a)$ with distinguished free variable a we shall write $A(m)$ for $A(a := \overline{m})$. We may refer to this as an assignment of a numeral to a variable.

Theorem 2.27. ***Embedding of*** $EA(I;O)$**.** *Assume that*

$$EA(I;O) \vdash \Gamma(x_0, \dots, x_l; a_0, \dots, a_k)$$

where all the free variables are displayed.

Then this derivation determines numbers $d, r \in \mathbf{N}$ *such that, for all* $n_0, \dots, n_l$ *and all* $m_0, \dots, m_k$*, if* $n \geq \max(n_0, \dots, n_l)$ *and* $m \geq \max(m_0, \dots, m_k)$ *then*

$$n; m \vdash^{\omega \cdot d}_{r} \Gamma(n_0, \dots, n_l; m_0, \dots, m_k).$$

Proof.

We proceed by induction on the height of the finite proof in $EA(I;O)$. Let $\vec{n} := n_0, \dots, n_l$ and $\vec{m} := m_0, \dots, m_k$.

1. Axioms. Assume that $EA(I;O) \vdash \Gamma(\vec{x}; \vec{a})$ where Γ contains any logical or arithmetic axiom. Then any instantiation of the free variables to numerals assures a true closed atom in Γ. Hence

$$n; m \vdash^{\omega \cdot 0}_{0} \Gamma(\vec{n}; \vec{m})$$

by the (L-Ax) rule of the infinitary calculus.

2. $(\vee)$. If $EA(I;O) \vdash \Gamma$ where the last rule applied is $(\vee)$ we have the premises

$$EA(I;O) \vdash \Gamma', A_i$$

for $i = 0$ or 1. Applying the induction hypothesis gives us

$$n; m \vdash^{\omega \cdot d_i}_{r} \Gamma', A_i$$

for $i = 0$ or 1. Now to apply the equivalent rule in the infinitary system we let $d :=$ max$(d_i) + 1$. Since $(\omega \cdot d)[n] = \omega \cdot d_i + (n+1)[n]$ we have $\omega \cdot d_i \in (\omega \cdot d)[n]$ and obtain

$$n; m \vdash^{\omega \cdot d}_{r} \Gamma.$$

3. $(\wedge)$. Now assume $EA(I;O) \vdash \Gamma$ where the last rule applied is $(\wedge)$. The premises are

$$EA(I;O) \vdash \Gamma', A_0 \qquad \text{and} \qquad EA(I;O) \vdash \Gamma', A_1.$$

We apply the induction hypothesis to give derivations of heights $\omega \cdot d_0$ and $\omega \cdot d_1$ in $EA_\infty(I;O)$. Then letting $d := \max(d_0, d_1) + 1$ we apply $(\wedge)$ in the infinitary system giving

$$n; m \vdash_r^{\omega \cdot d} \Gamma.$$

4. $(\forall)$. If we have a derivation in $EA(I;O)$ resulting from $(\forall)$ then the premise is, for b not free elsewhere in the sequent,

$$EA(I;O) \vdash \Gamma', A(b).$$

Applying the induction hypothesis for each possible assignment of k to the free output variable b gives

$$\{n; \max(m,k) \vdash_r^{\omega \cdot d'} \Gamma', A(k)\}_{k \in \mathbf{N}}.$$

We can therefore apply $(\forall)$ in the infinitary theory straight away with $d := d' + 1$. This gives

$$n; m \vdash_r^{\omega \cdot d} \Gamma, \forall a A(a).$$

5. $(\exists)$. This time the premise to the rule in $EA(I;O)$ is of the form

$$\vdash \Gamma', A(t(\vec{x}; \vec{a}))$$

where t is some term which may contain any number of the free variables $\vec{x}$ and $\vec{a}$ and possibly other variables $\vec{y}; \vec{b}$. In applying the induction hypothesis we assign these other variables the numeral 0. Therefore we find

$$n; m \vdash_{r'}^{\omega \cdot d_0} \Gamma', A(t(\vec{n}, \vec{m})). \tag{2.9}$$

In order to apply the existential rule in $EA_\infty(I;O)$ we must both compute the value of the term and substitute this value for $t(\vec{n}, \vec{m})$ in A. Let us assume that $t(\vec{n}, \vec{m})$ takes the value

k. That means $t(\vec{n},\vec{m}) = k$ is now a true atom in $EA_\infty(I;O)$. Hence using (L-Ax) gives $n;m \vdash^0_0 t(\vec{n},\vec{m}) = k$. Also using (L-Ax), by induction over the build-up of the formula A, we may show

$$n;m \vdash^{\omega\cdot d_1}_0 \neg A(t(\vec{n},\vec{m})), A(k) \tag{2.10}$$

for some $d_1 \in \mathbf{N}$ depending upon the complexity of A. We use these derivations to give

$$\dfrac{\overset{\text{(L-Ax)}}{n;m \vdash^0_0 t(\vec{n},\vec{m}) = k} \qquad \overset{\text{(2.10) and Weakening}}{n;m \vdash^{\omega\cdot d_1}_0 t(\vec{n},\vec{m}) \neq k, \neg A(t(\vec{n},\vec{m})), A(k)}}{n;m \vdash^{\omega\cdot(d_1+1)}_1 \neg A(t(\vec{n},\vec{m})), A(k)}\;\text{(L-Cut)}$$

From here we perform another cut with (2.9) on $A(t(\vec{n},\vec{m}))$. Letting $r := \max(r', |A|)$ and $d' : \max(d_0, d_1+1)+1$ this leaves

$$n;m \vdash^{\omega\cdot d'}_r \Gamma', A(k). \tag{2.11}$$

The existential rule may be re-applied once we have computed the value k using Lemma 2.17. Letting $d'' := \max(d', |t|+1)+1$ and $d := d''+1$ we have

$$\dfrac{\overset{\text{[Lemma 2.17]}}{n;m \Vdash^{\omega\cdot(|t|+1)} k} \qquad \overset{\overset{\overset{\text{(2.11)}}{n;m \vdash^{\omega\cdot d'}_r \Gamma', A(k)}}{\text{Weakening}}}{n;m \vdash^{\omega\cdot d''}_r \Gamma', A(k)}}{n;m \vdash^{\omega\cdot d}_r \Gamma, \exists a A(a)}\;(\exists)$$

Note how the conditions on tree-ordinals are satisfied since $\omega\cdot(|t|+1) \in (\omega\cdot d'')[n]$ and $\omega\cdot d'' \in (\omega\cdot d)[n]$.

6. (L-Cut). Where we have an (L-Cut) in $EA(I;O)$ the induction hypothesis will apply to the premises to give

$$n;m \vdash^{\omega\cdot d_0}_{r_0} \Gamma, \neg C \qquad n;m \vdash^{\omega\cdot d_1}_{r_1} \Gamma, C$$

where any extraneous variables have been assigned 0. We may re-apply the cut rule straight away with $d := \max(d_0, d_1)+1$ and $r := \max(r_0, r_1, |C|)$ to leave

$$n;m \vdash^{\omega\cdot d}_r \Gamma.$$

7. (P-Ind.). $EA_\infty(I;O)$ contains no induction rule so we look to replace the induction in $EA(I;O)$ by potentially ω-many cuts in EA_∞.

Assume that in $EA(I;O)$ we have a derivation of $\vdash \Gamma, A(x)$ and the last rule applied is induction. We must show that in $EA_\infty(I;O)$

$$n;m \vdash^{\omega\cdot d}_{r} \Gamma, A(k)$$

where importantly k is less than or equal to the input declaration n since it is an assignment on an input variable.

In $EA(I;O)$ the premises of such a derivation would have been

$$\vdash \Gamma', A(0) \qquad \vdash \Gamma', \neg A(a), A(a+1))$$

where a is not free elsewhere in Γ'. We apply the induction hypothesis to both of these to arrive at

$$n;m \vdash^{\omega\cdot d_0}_{r_0} \Gamma', A(0)$$

and for every $m' \in \mathbf{N}$

$$n;\max(m,m') \vdash^{\omega\cdot d_1}_{r_1} \Gamma', \neg A(m'), A(m'+1).$$

We look to apply (L-Cut) k times on the cut-formulae $A(0), A(1), \dots, A(k-1)$. Letting $d' := \max(d_0, d_1)$ and $r := \max(r_0, r_1, |A|)$ we will obtain, for any k:

$$n;\max(m,k) \vdash^{\omega\cdot d'+k}_{r} \Gamma, A(k). \tag{2.12}$$

The numeral k is to be an assignment to the free input x. Its value may be computed by Lemma 2.17 as

$$n;m \Vdash^{\omega} k. \tag{2.13}$$

Furthermore we know $k \leq n$. Hence by putting $d := d' + 1$ we have $\omega \cdot d' + k \in \omega \cdot d' + \omega[n] = \omega \cdot d[n]$ and $\omega \in \omega \cdot d[n]$. Applying the computation cut rule to (2.12) and

(2.13) in order to remove the declaration k will then give

$$n;m \vdash_r^{\omega \cdot d} \Gamma, A(k).$$

□

2.6 The Provably Recursive Functions of $EA(I;O)$

Definition 2.28. *A closed Σ_1-formula of the form*

$$\exists a_0, \ldots, \exists a_i B(a_0, \ldots, a_i)$$

where B is a bounded formula is said to be true at $w \in \mathbf{N}$ if there exist witnesses $w_0, \ldots, w_i \leq w$ such that $B(w_0, \ldots, w_i)$ is true in the standard model.

The property of Σ_1-persistence is that if a Σ_1-formula is true at w, it is also true at any $w' \geq w$. A set of Σ_1-formulae is true at w if at least one formula within the set is true at w. Bounded formulae which are true will be true for any witness w.

Lemma 2.29. *Let Δ be a set of Σ_1-formulae. Assume that in $EA_\infty(I;O)$ we have a derivation*

$$n;m \vdash_0^\alpha \Delta.$$

Further assume that $|t|$ is the maximum of the heights of any term t in Δ. Then Δ is true at $B_{\omega \cdot (|t|+1)+\alpha}(n;m)$.

Proof.

We use induction over the height of the derivation of Δ. In what follows we put $\beta' := \omega \cdot (|t| + 1) + \beta$ and $\alpha' := \omega \cdot (|t| + 1) + \alpha$. Then by Lemma 2.13 part 6 we find $B_\beta(n;m) \leq B_{\beta'}(n;m)$. Furthermore since $\beta \in \alpha[n]$ implies $\beta' \in \alpha'[n]$ we find by part 4 of the same lemma that if $\beta \in \alpha[n]$ then $B_{\beta'}(n;m) < B_{\beta'+1}(n;m) \leq B_{\alpha'}(n;m)$.

1. (L-Ax). If the derivation is an instance of (L-Ax) then Δ contains a true atom. Hence Δ is automatically true for any witness so it is certainly true at $B_{\alpha'}(n;m)$.

2. $(\vee)$. Where the last rule applied is $(\vee)$ we have the premise

$$n;m \vdash_0^\beta \Delta', A_i$$

for $i = 0$ or 1 and $\beta \in \alpha[n]$. Then inductively Δ', A_i is true at $B_{\beta'}(n;m)$. If it is Δ' that is true, then Δ will also be true at $B_{\beta'}(n;m)$. Otherwise, if A_i is true at $B_{\beta'}(n;m)$ then $A_0 \vee A_1$ and hence Δ will be true at $B_{\beta'}(n;m)$. In both cases, by persistence Δ is true at $B_{\alpha'}(n;m)$.

3. $(\wedge)$. The conjunction case gives us premises of the form

$$n;m \vdash_0^{\beta_0} \Delta', A_0 \qquad \text{and} \qquad n;m \vdash_0^{\beta_1} \Delta', A_1.$$

Inductively Δ', A_0 is true at $B_{\beta'_0}(n;m)$ and Δ', A_1 is true at $B_{\beta'_1}(n;m)$. If Δ' is true, then Δ will be true at $\max(B_{\beta'_0}(n;m), B_{\beta'_1}(n;m))$. Otherwise A_0 is true at $B_{\beta'_0}(n;m)$ and A_1 is true at $B_{\beta'_1}(n;m)$. Therefore $A_0 \wedge A_1$ is true at $\max(B_{\beta'_0}(n;m), B_{\beta'_1}(n;m))$. Either way, by persistence, Δ is true at $B_{\alpha'}(n;m)$.

4. $(\forall)$. The last rule applied could have been $(\forall)$ but since the formulae in question are Σ_1 the $\forall$ quantifier must be bounded. Letting Δ be $\Delta', \forall a(\neg a \leq t' \vee A(a))$ where A is a bounded formula we have for $\beta_k \in \alpha[n]$ the premises

$$\{n; \max(m,k) \vdash_0^{\beta_k} \Delta', \neg k \leq t' \vee A(k)\}_{k \in \mathbf{N}}.$$

By applying the induction hypothesis $\Delta', \neg k \leq t' \vee A(k)$ is true at $B_{\beta'_k}(n; \max(m,k))$ for every $k \in \mathbf{N}$.

Now consider two sub-cases. Firstly suppose that $\forall a(\neg a \leq t' \vee A(a))$ is true. Then as a true bounded formula it is automatically true at $B_{\alpha'}(n;m)$ and hence so is Δ. Otherwise we have that $\forall a(\neg a \leq t' \vee A(a))$ is false. Then there exists some $k \leq t'$ such that $A(k)$ is false. For this k, from the induction hypothesis, it must be the case that Δ' is true at $B_{\beta'_k}(n;\max(m,k))$ and hence Δ is true at $B_{\beta'_k}(n;\max(m,k))$. By Lemma 2.15 and Lemma 2.17, as $k \leq t'$, we find $k \leq B_{\omega\cdot(|t'|+1)}(n;m)$. Recall that β'_k is $\omega\cdot(|t|+1)+\beta_k$ and $|t|$ is the maximum of the heights of any term in Δ. Hence $k \leq B_{\beta'_k}(n;m)$ and

$$B_{\beta'_k}(n;\max(m,k)) \leq B_{\beta'_k}(n;\max(m,B_{\beta'_k}(n;m))) \leq B_{\beta'_k+1}(n;m) \leq B_{\alpha'}(n;m).$$

Therefore by persistence Δ is true at $B_{\alpha'}(n;m)$.

5. $(\exists)$. In the existential case, letting Δ be $\Delta', \exists a A(a)$, we have the premises

$$n;m \Vdash^{\beta_0} k \qquad n;m \vdash^{\beta_1}_0 \Delta', A(k).$$

The induction hypothesis applied to the right-hand premise implies that $\Delta', A(k)$ is true at $B_{\beta'_1}(n;m)$. If Δ' is true at $B_{\beta'_1}(n;m)$ then Δ is true at $B_{\beta'_1}(n;m)$ and by persistence Δ is true at $B_{\alpha'}(n;m)$.

Otherwise $A(k)$ is true at $B_{\beta'_1}(n;m)$. Then k is a witness for $\exists a A(a)$ so Δ will be true at $\max(k, B_{\beta'_1}(n;m))$. Using our bounding lemma (2.14) we know from the computation of k that $k \leq B_{\beta_0}(n;m)$. Hence Δ is true at $\max(B_{\beta_0}(n;m), B_{\beta'_1}(n;m))$. As $B_{\beta_0}(n;m) \leq B_{\alpha'}(n;m)$ and $B_{\beta'_1}(n;m) \leq B_{\alpha'}(n;m)$ we find by persistence that Δ is true at $B_{\alpha'}(n;m)$.

6. (C-Cut). From the premises

$$n;m \Vdash^{\beta_0} k \qquad n;k \vdash^{\beta_1}_0 \Delta$$

we inductively deduce that Δ is true at $B_{\beta'_1}(n;k)$. From the bounding lemma (2.14) we deduce from the left-hand premise that $k \leq B_{\beta_0}(n;m)$, hence $k \leq B_{\beta'_0}(n;m)$. Letting β' be the maximum of β'_0 and β'_1 we see that Δ is true at $B_{\beta'}(n;B_{\beta'}(n;m)) = B_{\beta'+1}(n;m)$. Then since $\beta' \in \alpha'[n]$, we find Δ is true at $B_{\alpha'}(n;m)$ by persistence.

$\square$

Theorem 2.30.

$$ProvRec(EA(I;O)) \subseteq \mathcal{E}^3$$

Proof.

Assume that a function f is provably recursive in $EA(I;O)$. Then we must have a derivation in $EA(I;O)$ of the existence condition for such a function to be provably recursive. That is,

$$EA(I;O) \vdash \exists b \exists c (C_f(\vec{x}, b, c))$$

where C is some bounded computational formula involving a computational code c and the output b of the function f applied to $\vec{x}$.

Applying the embedding theorem (2.27) assigning $\vec{x} := \vec{n}$ we find that $EA_\infty(I;O)$ will prove

$$n; 0 \vdash_r^{\omega \cdot d} \exists b \exists c (C_f(\vec{n}, b, c))$$

for some $d, r \in \mathbf{N}$ and $n :=$ max$(\vec{n})$.

Now appealing to Full Cut-Elimination 2.26 we find

$$n; 0 \vdash_0^{\alpha} \exists b \exists c (C_f(\vec{n}, b, c))$$

where $\alpha := 2_r(\omega \cdot d)$.

Then by Lemma 2.29 above we see that $\exists b \exists c (C_f(\vec{n}, b, c))$ is true at $B_{\alpha'}(n; 0)$ where α' is the exponential tree-ordinal $\omega \cdot (k+1) + 2_r(\omega \cdot d)$ for fixed $d, k, r \in \mathbf{N}$ and $n :=$ max$(\vec{n})$. By 2.12, $B_{\alpha'}(n; 0)$ is an elementary function. Hence $C_f(\vec{n}, m_0, m_1)$ is true when m_0 and m_1 are bounded by the elementary function $B_{\alpha'}(n; 0)$. The function $f(\vec{n})$ may be computed by finding the least pair m_0 and m_1 such that $C_f(\vec{n}, m_0, m_1)$ is true. The formula C_f involves only bounded quantifiers and propositional connectives and the bounds on finding m_0 and m_1 are elementary. Hence $f(\vec{n})$ is elementarily definable from the elementary function $B_{\alpha'}(n; 0)$. Therefore $f(\vec{n})$ itself is an elementary function.

□

Corollary 2.31.

$$ProvRec(EA(I;O)) = \mathcal{E}^3$$

Proof.

The left to right inclusion is given above whilst the right to left inclusion is Theorem 1.24 in Chapter 1.

□

Chapter 3

EA^1 - A Conservative Closure of *EA(I;O)*

3.1 Introduction and Definitions for EA^1

This chapter begins an investigation into how we might extend the strength of $EA(I;O)$ so that we arrive at theories whose provably recursive functions belong to higher levels of the Grzegorczyk Hierarchy. The approach we explore is to define a hierarchy of theories where each successive theory is strictly layered upon the preceding one. To this end we begin by defining a new theory layered over $EA(I;O)$ which has more natural substitution properties by collapsing the distinction between input and output variables.

Definition 3.1. ***The theory*** EA^1**.**

The language of EA^1 contains just one type of variable, which we call output variables, using the notation of Chapter 1: $a, b, c, a_0, \vec{a}, \ldots$. The rest of the language is the same as that of $EA(I;O)$. The definitions of terms and formulae are standard and we again adopt a Tait-style sequent calculus. We have axioms for excluded middle and equality, and the standard rules for disjunction, conjunction, existential quantification, universal quantification and cut. All the arithmetic axioms listed in Chapter 1 are included. There is no induction rule in EA^1. We do however give the theory additional strength by including a *closure axiom*. It allows certain end sequents of derivations in $EA(I;O)$ to be taken as axioms in EA^1 on two provisos. Firstly we require that the sequent is a Σ_1-formula. Secondly we drop the distinction between input and output variables in the sequent.

The Σ_1**-closure axiom** of EA^1 reads,

$$\text{(C-Ax)} \qquad EA^1 \vdash \Gamma(\vec{c}), A(\vec{a}, \vec{b}) \qquad \text{if} \quad EA(I;O) \vdash A(\vec{x}; \vec{b}).$$

where A is a Σ_1-formula, Γ is an arbitrary set of formulae, and where all the free variables of Γ, A are indicated (with $\vec{a}$, $\vec{b}$ and $\vec{c}$ disjoint).

Remarks

The (C-Ax) rule replaces the uninterpreted (arbitrary) input constants $\vec{x}$ in the original $EA(I;O)$ derivation by 'proper' variables in EA^1. The change of symbols to $\vec{a}$ is intended to indicate that a collapse of the variable separation has taken place. Following the application of the rule we may apply universal quantification to the variables $\vec{a}$. Thus (C-Ax) resembles a formalized ω-rule in EA^1 for Σ_1-formulae. We shall discuss this further at the conclusion of this section.

Note that derivations such as $EA(I;O) \vdash Prog_a A(a) \to A(x)$ cannot be carried across (C-Ax) unless A is a bounded formula. Hence we may deduce a form of induction by showing $EA^1 \vdash Prog_a A(a) \to \forall a A(a)$ but only for bounded formulae. Since our term structure is still limited to sub-elementary functions, bounded induction does not increase the strength of the theory. If we allowed formulae of higher complexity to pass through (C-Ax) we would find functions of much greater complexity than the elementary functions to be provably recursive.

By including a set of arbitrary side-formulae $\Gamma(\vec{c})$ we ensure that EA^1 admits weakening.

It should be clear that the basic results (Lemma 1.8, Lemma 1.9 as well as Lemma 1.17) from Chapter 1 are provable in EA^1.

Definition 3.2. *A function f is said to be* ***provably recursive*** *in the theory EA^1 if its graph can be defined by a Σ_1-formula $\exists c C_f(\vec{a}, b, c)$, where C_f is Δ_0, such that*

$$EA^1 \vdash \forall \vec{a} \exists b \exists c (C_f(\vec{a}, b, c))$$

$$EA^1 \vdash \forall \vec{a} \forall b \forall b' (\exists c (C_f(\vec{a}, b, c)) \wedge \exists c (C_f(\vec{a}, b', c)) \rightarrow b = b').$$

We denote the class of functions provably recursive in EA^1 by $ProvRec(EA^1)$.

Note how we now have two subtly different notions of provably recursive: one for two-sorted theories such as $EA(I;O)$ (Definition 1.15) and the one given above for a single-sorted theory like EA^1. The former defines a function as provably recursive for any arbitrary input constants $\vec{x}$. The latter now defines a function as provably recursive *for all* variables $\vec{a}$ from within the theory via a universal quantifier. This matches the usual Π_2 definition of a provably recursive function.

Proposition 3.3. *If the function f is provably recursive in $EA(I;O)$ then f is also provably recursive in EA^1. Hence the elementary functions are provably recursive in EA^1.*

Proof.

For any provably recursive function in $EA(I;O)$ we have for some computational formula $C_f(\vec{x}; b, c)$

$$EA(I;O) \vdash \exists b \exists c (C_f(\vec{x}; b, c)) \tag{3.1}$$

and

$$EA(I;O) \vdash \forall b \forall b' (\exists c (C_f(\vec{x}; b, c)) \wedge \exists c (C_f(\vec{x}; b', c)) \rightarrow b = b'). \tag{3.2}$$

From (3.1) we may immediately apply (C-Ax) and universal quantification to give the existence condition

$$EA^1 \vdash \forall \vec{a} \exists b \exists c (C_f(\vec{a}, b, c)).$$

Using (3.2) we deduce in $EA(I;O)$ by inversions and disjunction

$$\neg C_f(\vec{x}; b, c) \vee \neg C_f(\vec{x}; b', c') \vee b = b'.$$

Now that we have a Σ_1-formula, applying (C-Ax) gives us

$$EA^1 \vdash \neg C_f(\vec{a}; b, c) \vee \neg C_f(\vec{a}; b', c') \vee b = b'.$$

Applying quantifications yields the uniqueness requirement:

$$EA^1 \vdash \forall \vec{a} \forall b \forall b' (\exists c (C_f(\vec{a}, b, c)) \wedge \exists c (C_f(\vec{a}, b', c)) \rightarrow b = b').$$

Hence f is a provably recursive function in EA^1.

$\square$

As Wirz notes in [68], "It is a common criticism that $EA(I;O)$ doesn't provide a direct mechanism for substitute terms for input variables. That is, its provably total functions aren't intensionally closed under (predicative) composition." Of course, we know the provably recursive functions are extensionally closed under composition since the elementary functions are. Wirz [68] was able to circumvent this problem by generalizing the approaches of Ostrin and Wainer [48] and deducing an input substitution rule for any provably recursive function. In EA^1 we may use (C-Ax) to quantify over variables which were fixed as free input constants in a particular $EA(I;O)$ derivation. Using logical cuts and the coding machinery we may then easily compose provably recursive functions. As a result we argue that EA^1 may appeal as a more *natural* theory to work in whilst remaining, in terms of provably recursive functions, conservative over $EA(I;O)$. The following theorem motivates our view of EA^1 as a theory of closure over $EA(I;O)$.

Theorem 3.4. *$ProvRec(EA^1)$ is closed under Composition.*

That is, for $j, k > 0$ if the j-ary function h and the k-ary functions $g_1, \dots, g_j$ are all provably recursive in EA^1 then for $\vec{a} := a_1, \dots, a_k$ the composition function f defined as

$$f(\vec{a}) := h(g_1(\vec{a}), \dots, g_j(\vec{a}))$$

is also provably recursive in EA^1.

Proof.

Without loss of generality let $j := 2$. We may assume that we are given suitable computational formulae C_{g_1}, C_{g_2}, C_h for each of the functions g_1, g_2 and h. Then define the computational formula for the composition function f as

$$\begin{aligned} C_f(\vec{a}, b, c) \quad :\equiv \quad & lh(c) = 3 \wedge (c)_0 \neq 0 \wedge (c)_1 \neq 0 \wedge (c)_2 \neq 0 \wedge l((c)_2) = b \\ & \wedge C_{g_1}(\vec{a}, l((c)_0), r((c)_0)) \wedge C_{g_2}(\vec{a}, l((c)_1), r((c)_1)) \\ & \wedge C_h(l((c)_0), l((c)_1), l((c)_2), r((c)_2)). \end{aligned} \tag{3.3}$$

Then clearly $f(\vec{n}) = m$ if and only if $\exists c C_f(\vec{n}, m, c)$ is true.

We begin by deriving the existence condition for f to be provably recursive in EA^1. Let t be the term $\langle p(b_1, d_1), p(b_2, d_2), p(b, d)\rangle$. Then from the definition of C_f, using the coding axioms, it is straightforward to derive in EA^1

$$\neg C_{g_1}(\vec{a}, b_1, d_1), \neg C_{g_2}(\vec{a}, b_2, d_2), \neg C_h(b_1, b_2, b, d), C_f(\vec{a}, b, t).$$

Hence by quantification

$$\begin{aligned} & \big(\forall \vec{a} \exists b_1 \exists d_1 C_{g_1}(\vec{a}, b_1, d_1) \wedge \forall \vec{a} \exists b_2 \exists d_2 C_{g_2}(\vec{a}, b_2, d_2) \wedge \forall b_1 \forall b_2 \exists b \exists d C_h(b_1, b_2, b, d)\big) \\ & \qquad \rightarrow \forall \vec{a} \exists b \exists c C_f(\vec{a}, b, c). \end{aligned} \tag{3.4}$$

Appealing to the assumptions that g_1, g_2 and h are provably recursive in EA^1, we have from the existence conditions, a derivation of each of the three conjuncts in the antecedent of (3.4). Hence by three uses of cut we have the existence condition for f:

$$EA^1 \vdash \forall \vec{a} \exists b \exists c C_f(\vec{a}, b, c).$$

The uniqueness condition is also easily derived and we argue informally. Let us assume firstly that we are given a c such that $C_f(\vec{a}, b, c)$ and secondly that we are given a c' such that $C_f(\vec{a}, b', c')$ in order to show $b = b'$. From the definition of C_f we find

$C_{g_1}(\vec{a}, l((c)_0), r((c)_0))$ by the first assumption and $C_{g_1}(\vec{a}, l((c')_0), r((c')_0))$ by the second. Since the uniqueness property holds for g_1 we deduce $l((c)_0) = l((c')_0)$. Likewise, with respect to g_2, we see $l((c)_1) = l((c')_1)$. Now by substitution on the uniqueness of h we may deduce $l((c)_2) = l((c')_2)$. But from our first assumption $l((c)_2) = b$ and from our second assumption $l((c')_2) = b'$ hence $b = b'$.

□

Remarks

We conclude this section with some brief remarks on a semantic interpretation of (C-Ax) and on predicativity. Although not a central concern of this thesis, a model-theoretic consideration of the (C-Ax) rule is useful here. One imagines constructing a model $\mathcal{M}$ of $EA(I;O)$ by starting with a structure $\langle \mathbb{O}, 0^{\mathcal{M}}, +1^{\mathcal{M}}, +^{\mathcal{M}}, \cdot^{\mathcal{M}}, \ldots \rangle$ and adjoining a set of *input constants* $\mathbb{I} := \{i_0, i_1, \ldots, i_k, \ldots\}$ assigned to the inputs x_j. When the values assigned to the x_j are arbitrary values *in* $\mathbb{O}$, the closure axiom becomes a valid principle.

We claimed in the introduction that $EA(I;O)$ might be regarded as predicative in Nelson's sense [42] because induction up to input 'numbers' avoids impredicative quantification over all such numbers. We informally argue that EA^1 may be seen as a impredicative extension of $EA(I;O)$ by seeing (C-Ax) as a type of internalised ω rule: from $A(x)$ for each $x := i_k$ we may derive $\forall a A(a)$. It is clear that EA^1 does not satisfy Nelson's requirements for a predicative arithmetic since the exponential function is provably *total* within the theory. In fact our (C-Ax) rule, in collapsing the variable separation of $EA(I;O)$, mirrors what Nelson objects to in his remarks on the impredicativity of standard theories of arithmetic, p.14 [42]. In $EA(I;O)$, if $A(a)$ is progressive in a we may conclude $A(x)$. We may interpret this along Nelson's lines as 'if n is a number then $A(n)$.' In EA^1 we make the impredicative step in (C-Ax) to claim $\forall a A(a)$ or 'for all numbers $A(n)$.'

For a given $EA(I;O)$ derivation of a Σ_1-formula $A(\vec{x})$ (such as that defining a provably recursive function) the closure axiom followed by universal quantification gives in EA^1 a derivation of $\forall\vec{a}A(\vec{a})$. The universal quantifiers $\forall\vec{a}$ are thus to be regarded as quantifiers ‘with computational content’ in the sense of Schwichtenberg [56] (originating from the work of Berger [8]).

3.2 Bounded Arithmetic and EA^1

We have already alluded to the fact that EA^1 admits bounded induction. This leads to simple correspondence between bounded arithmetic and EA^1.

$I\Delta_0$ is the sub-theory of Peano Arithmetic in which induction is restricted to only apply to bounded formulae. In [23] it is shown that in $I\Delta_0$ there exists a Δ_0 formula, we call $\exp(a, b)$, which defines the graph of exponentiation, $2^a = b$. But by Parikh’s Theorem [49] one cannot show that the exponential function is provably recursive since b cannot be bounded by a term in the language. $I\Delta_0 + \exp$ is the extension of $I\Delta_0$ obtained by adding an axiom asserting that the exponential function is total. The system is sufficient to prove a number of important results in number theory and has many equivalent formulations in all of which the elementary functions are provably recursive, see for example [12], [30] or [65].

We present $I\Delta_0 + \exp$ using a Tait-style sequent calculus giving the induction axiom schema as a rule of inference. The logical symbols are the same as EA^1. The non-logical symbols are the constant symbol 0 and the function symbols $+1, +, \cdot$. The relation $\leq$ is defined by $a \leq b :\equiv \exists c(a + c = b)$. The logical axioms rules governing $\vee, \wedge, \exists, \forall$ and Cut are standard.

Definition 3.5. ***The non-logical axioms of*** $I\Delta_0 + \exp$**.**

$I\Delta_0 + \exp$ includes the usual equality axioms as well as (universal closures of) the following arithmetic axioms (with arbitrary side-formulae Γ omitted)

$$a + 1 \neq 0 \tag{3.5}$$

$$a + 1 = b + 1 \rightarrow a = b \tag{3.6}$$

$$a = 0 \vee \exists b(b + 1 = a) \tag{3.7}$$

$$a + 0 = a \tag{3.8}$$

$$a + (b + 1) = (a + b) + 1 \tag{3.9}$$

$$a \cdot 0 = 0 \tag{3.10}$$

$$a \cdot (b + 1) = a \cdot b + a \tag{3.11}$$

Letting $\exp(a, b)$ be some Δ_0 formula expressing the fact that $2^a = b$, for arbitrary side-formulae Γ we add the axiom

$$\Gamma, \forall a \exists b \exp(a, b) \tag{3.12}$$

The **bounded induction** rule is, for any $B \in \Delta_0$ and arbitrary set of formulae Γ where a is not free in Γ:

(Δ_0-Ind.)
$$\frac{\Gamma, B(0) \qquad \Gamma, \neg B(a), B(a+1)}{\Gamma, \forall a B(a).}$$

Proposition 3.6. ***Embedding $I\Delta_0$ into EA^1.***

$$I\Delta_0 + \exp \vdash \Gamma \qquad \Rightarrow \qquad EA^1 \vdash \Gamma.$$

Proof.

We use induction over the derivation of Γ inside $I\Delta_0 + \exp$ with a case distinction according to the last rule applied.

1. For the basis of the induction firstly assume that $I\Delta_0 + \exp \vdash \Gamma$ where Γ contains a logical, equality or arithmetic axiom other than (exp). Then as all such axioms are available to us in EA^1 clearly $EA^1 \vdash \Gamma$.

If the derivation is the axiom $\Gamma, \forall a \exists b \exp(a, b)$ we may prove directly an appropriate translation in EA^1. Recall from Chapter 1, $EA(I;O) \vdash \exists b \exists c E(x, b, c)$ where $E(a, b, c)$

is a bounded formula expressing that c is a sequence code for the computation $2^x = b$. Then by (C-Ax), and universal quantification we have $EA^1 \vdash \Gamma, \forall a \exists b \exists c E(a,b,c)$. We take this derivation to be the EA^1 translation of the $I\Delta_0 + \exp$ axiom where $\exp(a,b)$ is replaced by the Σ_1-formula $\exists c E(a,b,c)$.

2. It is clear that for the rules of inference $\vee, \wedge, \exists, \forall$ and Cut, we may take the premises of the rule in $I\Delta_0 + \exp$, apply the induction hypothesis and then re-apply the same rule in EA^1.

3. Assume the derivation results from an application of (Δ_0-Ind.) with $\Gamma :\equiv \Gamma', \forall a B(a)$ where B is a bounded formula. Then from the premises $\Gamma', B(0)$ and $\Gamma', \neg B(a), B(a+1)$ we apply the induction hypothesis to giving derivations of the same sequents in EA^1. Then by universal quantification and conjunction

$$EA^1 \vdash \Gamma', B(0) \wedge \forall a(B(a) \rightarrow B(a+1)). \tag{3.13}$$

From the predicative induction rule in $EA(I;O)$ we may show

$$EA(I;O) \vdash (B(0) \wedge \forall a(B(a) \rightarrow B(a+1)) \rightarrow B(x).$$

Then as B is a bounded formula this sequent is a Σ_1-formula. We apply (C-Ax) to yield, for b not free in Γ',

$$EA^1 \vdash \Gamma', (B(0) \wedge \forall a(B(a) \rightarrow B(a+1)) \rightarrow B(b).$$

Applying a cut with (3.13) followed by universal quantification leaves the required result:

$$EA^1 \vdash \Gamma', \forall a B(a).$$

□

This result gives as a corollary another proof of Proposition 3.3 since the elementary functions are provably recursive in $I\Delta_0 + \exp$.

3.3 An Infinitary Theory for EA^1

It still remains to show that EA^1 is indeed a conservation extension of $EA(I;O)$ in terms of its provably recursive functions. To this end we need to eliminate cuts from EA^1 derivations. This is straightforward except where (C-Ax) is used since we would we need to assume the original $EA(I;O)$ derivation is cut-free. However this can be achieved in the setting of $EA_\infty(I;O)$. Hence we shall develop an infinitary theory for EA^1 which via a closure axiom incorporates derivations from $EA_\infty(I;O)$. Then following the methods of Chapter 2 we look to embed EA^1 in this new infinitary theory and deduce that the provably recursive functions of EA^1 are at most the elementary functions.

Definition 3.7. ***The infinitary theory*** EA^1_∞.

EA^1_∞ is based upon the approach used for $EA_\infty(I;O)$ in the previous chapter. The proof gate carries an elementary tree-ordinal α and also a finite parameter d to measure height written d, α. We may view the height of a derivation as the 'composition' $d \circ \alpha$. This is because, other than its introduction in the closure axiom, α remains fixed in the rules of inference whilst d increases. Hence a derivation in EA^1_∞ is seen as one of height α followed by one of height d.

As there is only one type of variable in EA^1 we only require one natural number declaration to the left of the proof gate from a single assumed domain of numbers. Thus sequents in EA^1_∞ take the form

$$m \vdash^{d,\alpha}_r \Gamma.$$

We read this as 'given fixed natural number values $\leq m$ from the output domain, the truth of Γ can be established in ($d \circ \alpha$)-many steps using cuts on formulae whose rank is $\leq r$.' These sequents are defined inductively in the following.

Logical Rules

(C-Ax) $$\max(n_0, m_0) \vdash_0^{d,\alpha} \Gamma$$ for any d, if in $EA_\infty(I;O)$ we already have $n_0; m_0 \vdash_0^{\alpha} \Gamma$.

(L-Ax) $$m \vdash_r^{d,\alpha} \Gamma$$ for any d, α and r, if Γ contains a true atom.

($\vee$) $$\frac{m \vdash_{r_i}^{d_i,\alpha} \Gamma, A_i}{m \vdash_r^{d,\alpha} \Gamma, A_0 \vee A_1}$$ if $d_i < d$ and $r_i \leq r$ where $i = 0$ or 1.

($\wedge$) $$\frac{m \vdash_{r_0}^{d_0,\alpha} \Gamma, A_0 \qquad m \vdash_{r_1}^{d_1,\alpha} \Gamma, A_1}{m \vdash_r^{d,\alpha} \Gamma, A_0 \wedge A_1}$$ if $d_0, d_1 < d$ and $r_0, r_1 \leq r$.

($\forall$) $$\frac{\{\max(m,k) \vdash_{r'}^{d_k,\alpha} \Gamma, A(k)\}_{k \in \mathbb{N}}}{m \vdash_r^{d,\alpha} \Gamma, \forall a A(a)}$$ if for all $k \in \mathbf{N}$, $d_k < d$ and $r' \leq r$.

($\exists$) $$\frac{m \Vdash^{d_0,\alpha} k \qquad m \vdash_{r_1}^{d_1,\alpha} \Gamma, A(k)}{m \vdash_r^{d,\alpha} \Gamma, \exists a A(a)}$$ if $d_0 < d_1 < d$ and $r_1 \leq r$.

(L-Cut) $$\frac{m \vdash_{r_0}^{d_0,\alpha} \Gamma, \neg C \qquad m \vdash_{r_1}^{d_1,\alpha} \Gamma, C}{m \vdash_r^{d,\alpha} \Gamma}$$ if $d_0, d_1 < d$ and $|C|, r_0, r_1 \leq r$.

(C-Cut) $$\frac{m \Vdash^{d_0,\alpha} k \qquad k \vdash_{r_1}^{d_1,\alpha} \Gamma}{m \vdash_r^{d,\alpha} \Gamma}$$ if $d_0, d_1 < d$ and $r_1 \leq r$.

Computational Rules

($\mathbb{O}$-Ax) $$\max(n_0, m_0) \Vdash^{d,\alpha} k$$ for any d, if $k \leq B_\alpha(n_0; m_0)$.

($\mathbb{O}$-Cut) $$\frac{m \Vdash^{d_0,\alpha} m' \qquad m' \Vdash^{d_1,\alpha} k}{m \Vdash^{d,\alpha} k}$$ if $d_0, d_1 \leq d$.

Remarks

The (C-Ax) rule plays the same role as the corresponding rule in EA^1 allowing derivations from $EA_\infty(I;O)$ to be brought in as axioms. Such derivations must be cut-free so that EA^1_∞ possesses cut-elimination. However we have no need for any restriction on the complexity of the formulae in Γ. This is possible since our interest will ultimately be in cut-free derivations of Σ_1-sets. Following uses of (C-Ax), α remains fixed throughout applications of other EA^1_∞ rules.

Aside from the (C-Ax) rule we have essentially the same rules as $EA_\infty(I;O)$. We may now allow a finite number d to control the height of derivations involving these rules. This is because there are no inductions in EA^1 which must be embedded in EA^1_∞ and hence no diagonalisation across the parameter m.

The computation rules are also set up to layer on top of those in $EA_\infty(I;O)$. We have an axiom ensuring access to a computation of any numeral less than $B_\alpha(n_0;m_0)$ and a cut rule enabling composition. Again the measure d is finite since we need not diagonalise. Cut-elimination will involve only a finite increase in d thus the bounding function required in EA^1_∞ will only require finite compositions on output values.

3.4 Computations in EA^1_∞

Recall the definition of the bounding function $B_\alpha(n;m)$ given in 2.11 on page 46. In order to define a suitable bounding function for EA^1_∞ we must consider finite iterations of $B_\alpha(n;m)$.

Definition 3.8. *For* $\alpha \in \Omega^S$ *and fixed* $d \in \mathbf{N}$ *the functions* $B_{d,\alpha} : \mathbf{N} \to \mathbf{N}$ *are defined by recursion over* d*:*

$$\begin{aligned} B_{0,\alpha}(m) &:= B_\alpha(m;m), \\ B_{d+1,\alpha}(m) &:= B_{d,\alpha}(B_{d,\alpha}(m)). \end{aligned}$$

From the definition we see $B_{d,\alpha}(m)$ iterates 2^d-times, on both arguments, the function $B_\alpha(m;m)$.

Corollary 3.9. *Given $\alpha \in E(\omega)$ and $d \in \mathbf{N}$ the function $B_{d,\alpha} : \mathbf{N} \to \mathbf{N}$ is elementary.*

Proof.

In Lemma 2.12 of Chapter 2 we determined that $B_\alpha(n;m)$ is an elementary function in n, m for any elementary tree-ordinal α. Hence the one place function, $B_{0,\alpha}(m)$, is an elementary function in m. If $d \in \mathbf{N}$ is fixed, $B_{d,\alpha}(m)$ is a finite number of compositions of an elementary function. Therefore it is also an elementary function in m.

$\square$

Lemma 3.10. *For any $\alpha \in \Omega^S$ and $d, m \in \mathbf{N}$, $B_{d,\alpha}(m)$ is strictly increasing in m and d.*

Proof.

Recall the majorization properties for $B_\alpha(n;m)$ given in Lemma 2.13 on page 47. Since $B_\alpha(n;m)$ is strictly increasing in m and increasing in n we find $B_{d,\alpha}(m)$ is strictly increasing in m by a simple induction on d. Furthermore, $B_\alpha(n;m)$ is strictly positive in m. Hence $B_{d,\alpha}(m)$ is strictly increasing in d, again using induction on d.

$\square$

Lemma 3.11. ***Bounding for** EA^1_∞.*

$$m \Vdash^{d,\alpha} k \qquad \Leftrightarrow \qquad k \leq B_{d,\alpha}(m).$$

Proof.

We use induction over the height of the computation of k. If the computation is an ($\mathbb{O}$-Ax) then $k \leq B_\alpha(n_0; m_0)$ where $\max(n_0, m_0) = m$. Hence, as $B_\alpha(n_0; m_0) \leq B_\alpha(m;m) \leq B_{d,\alpha}(m)$ for any d by the previous lemma, the result follows. Else, the computation results from ($\mathbb{O}$-Cut) and the induction hypothesis applied to the premises of the rule gives $k \leq B_{d_0,\alpha}(m')$ and $m' \leq B_{d_1,\alpha}(m)$ where $d_0, d_1 < d$. Let $d' := \max(d_0, d_1)$. Then the lemma above immediately gives $k \leq B_{d',\alpha}(B_{d',\alpha}(m)) \leq B_{d,\alpha}(m)$.

For the right to left implication we use induction over d. If $d = 0$ we simply use ($\mathbb{O}$-Ax) as $k \leq B_{0,\alpha}(m) = B_\alpha(m; m)$. If $d = d_0 + 1$ then letting $m' := B_{d_0,\alpha}(m)$ we have $k \leq B_{d_0,\alpha}(m')$ and $m' \leq B_{d_0,\alpha}(m)$. Applying the induction hypothesis allows the use of the ($\mathbb{O}$-Cut) rule to obtain the required result.

□

Lemma 3.12. ***Weakening for Computations.*** *If we have the computation*

$$m \Vdash^{d,\alpha} k$$

and if $m \leq m', k \geq k'$ *and* $d \leq d', \alpha[m] \subseteq \alpha'[m]$ *then for any* $\gamma \in \Omega^S$ *then we also have the computation*

$$m' \Vdash^{d',\gamma+\alpha'} k'.$$

Proof.
We use induction over the height of the computation of k. Assume that the computation is an ($\mathbb{O}$-Ax). Hence $k \leq B_\alpha(n_0; m_0)$ where $\max(n_0, m_0) = m$. Then from the proof of 2.15 we know $k' \leq k \leq B_\alpha(n_0; m_0) \leq B_\alpha(m; m) \leq B_{\gamma+\alpha'}(m'; m')$. The result will follow by another ($\mathbb{O}$-Ax).

Now assume the computation results from an application of ($\mathbb{O}$-Cut). We have premises of the form

$$m \Vdash^{d_0,\alpha} m_0 \qquad m_0 \Vdash^{d_1,\alpha} k.$$

We apply the induction hypothesis to obtain

$$m' \Vdash^{d_0,\gamma+\alpha'} m_0 \qquad m_0 \Vdash^{d_1,\gamma+\alpha'} k'.$$

Hence applying ($\mathbb{O}$-Cut) again will give $m' \Vdash^{d',\gamma+\alpha'} k'$.

□

Recall the definition of the height of term t notated by $|t|$ in 2.16.

Lemma 3.13. *For any term $t(\vec{a})$ in EA^1 and any number m, if the value of the term upon substituting $m_i \leq m$ for each a_i is the number k then in EA^1_∞*

$$m \Vdash^{0,\omega\cdot(|t|+1)} k.$$

Proof.

Directly from the corresponding Lemma 2.17 in Chapter 2, $k \leq B_{\omega\cdot(|t|+1)}(m;m)$. The result then follows from ($\mathbb{O}$-Ax).

□

3.5 Cut-Elimination for EA^1_∞

The bulk of the work for the results in this section will follow the approach of the corresponding results in Chapter 2 for $EA_\infty(I;O)$ since the lemmas are essentially the same. Indeed matters are made easier here since, except in the (C-Ax) rule, only the finite measure d increases as EA^1_∞ rules are applied. We deal with cases of (C-Ax) by appealing to the analogous results for $EA_\infty(I;O)$.

Lemma 3.14. ***Weakening for Logical Rules.*** *If we have a derivation*

$$m \vdash^{d,\alpha}_{r} \Gamma$$

and if $m \leq m', \Gamma \subseteq \Gamma'$ and $d \leq d', \alpha[m] \subseteq \alpha'[m]$ then for any $\gamma \in \Omega^S$ we also have the derivation

$$m' \vdash^{d',\gamma+\alpha'}_{r} \Gamma'.$$

Proof.

Using induction on the height of the derivation let us first assume that the sequent in question is an instance of (C-Ax). Then we have an $EA_\infty(I;O)$ derivation of $n_0; m_0 \vdash_0^\alpha \Gamma$ where $\max(n_0, m_0) = m$. Applying weakening in $EA_\infty(I;O)$ using Lemma 2.18 gives $m'; m' \vdash_0^{\gamma+\alpha'} \Gamma'$. But then by (C-Ax) in EA^1_∞ we arrive at $m' \vdash_0^{d',\gamma+\alpha'} \Gamma'$.

If the sequent is a logical axiom then the result is given immediately as another logical axiom. The remaining cases are also straightforward. For each rule of EA^1_∞ we apply the induction hypothesis to the premises of the rule and look to re-apply the rule. Where we have a computational sequent as a premise we appeal to computational weakening using Lemma 3.12 above. For example consider a use of the existential rule. The premises in this case would be of the form

$$m \Vdash^{d_0,\alpha} k \qquad m \vdash_r^{d_1,\alpha} \Gamma, A(k)$$

where $d_0 < d_1 < d$. Using the induction hypothesis on the right and 3.12 on the left gives

$$m' \Vdash^{d_0,\gamma+\alpha'} k \qquad m' \vdash_r^{d_1,\gamma+\alpha'} \Gamma', A(k)$$

from which we just re-apply the rule with $d_0 < d_1 < d'$.

□

Lemma 3.15. ***Inversions.***

$$m \vdash_r^{d,\alpha} \Gamma, A_0 \wedge A_1 \quad \Rightarrow \quad m \vdash_r^{d,\alpha} \Gamma, A_i \qquad \textit{where } i = 0 \textit{ or } 1.$$

$$m \vdash_r^{d,\alpha} \Gamma, \forall a A(a) \quad \Rightarrow \quad \max(m,k) \vdash_r^{d,\alpha} \Gamma, A(k) \quad \textit{for every } k \in \mathbf{N}.$$

Proof.

We use induction over the height of the derivation in both cases.

1. If the derivation is a (C-Ax) then we have a derivation in $EA_\infty(I;O)$ of $n_0; m_0 \vdash_0^\alpha \Gamma, A_0 \wedge A_1$ for $\max(n_0, m_0) = m$. Thus by applying conjunction inversion in $EA_\infty(I;O)$ we may use (C-Ax) again to provide a derivation in EA^1_∞ of $m \vdash_0^{d,\alpha} \Gamma, A_i$ for $i = 0$ or 1. The (L-Ax) case is trivial. If the last rule applied were conjunction and $A_0 \wedge A_1$ is

principal we apply the induction hypothesis to the premises and use weakening. All other cases are straightforward by the induction hypothesis.

2. The (C-Ax) case follows similarly to that above by making use of the corresponding universal inversion lemma (2.20) in $EA_\infty(I;O)$. If $m \vdash_r^{d,\alpha} \Gamma, \forall a A(a)$ is an instance of (L-Ax) then clearly so is $\max(m,k) \vdash_r^{d,\alpha} \Gamma, A(k)$ for every $k \in \mathbf{N}$. Now assume universal quantification is the last rule applied where $\forall a A(a)$ is the principal formula. We have premises of the form $\max(m,k) \vdash^{d_k,\alpha} \Gamma, \forall a A(a), A(k)$ for each $k \in \mathbf{N}$ where $d_k < d$. Applying the induction hypothesis and then weakening the bound d_k to d provides the required result. The remaining cases follow the usual pattern. For example consider an application of the existential quantification rule from premises

$$m \Vdash^{d_0,\alpha} k' \qquad m \vdash_r^{d_1,\alpha} \Gamma, B(k'), \forall a A(a)$$

where $d_0 < d_1 < d$. We require the induction hypothesis on the right and computational weakening on the left to obtain

$$\max(m,k) \Vdash^{d_0,\alpha} k' \qquad \max(m,k) \vdash_r^{d_1,\alpha} \Gamma, B(k'), A(k)$$

for every $k \in \mathbf{N}$. The result follows by re-applying the rule.

□

Lemma 3.16. ***False Atom.*** *If C is atomic and true so that $\neg C$ is false then*

$$m \vdash_r^{d,\alpha} \Gamma, \neg C \quad \Rightarrow \quad m \vdash_r^{d,\alpha} \Gamma.$$

Proof.

If the derivation of $\Gamma, \neg C$ is an instance of (C-Ax) then in $EA_\infty(I;O)$ we have $n_0; m_0 \vdash_0^\alpha \Gamma, \neg C$ for $\max(n_0, m_0) = m$. By the false atom lemma for $EA_\infty(I;O)$ (2.23) we obtain $n_0; m_0 \vdash_0^\alpha \Gamma$ and the result follows by (C-Ax) again. For all other cases we may simply follow the lines of the corresponding lemma for $EA_\infty(I;O)$.

□

Lemma 3.17. ***Cut-Rank Reduction.*** *Assume in* EA^1_∞ *we have*

$$m \vdash^{d,\alpha}_r \Gamma_0, C \quad \textit{and} \quad m \vdash^{d',\alpha}_r \Gamma_1, \neg C$$

where C *is either an atom or of the form* $C_0 \vee C_1$ *or* $\exists a C_0(a)$ *with* $|C|$*=r+1. Then*

$$m \vdash^{d'+d,\alpha}_r \Gamma_0, \Gamma_1.$$

Proof.

The proof proceeds by induction over the height of the derivation of Γ_0, C with cases according to the last rule applied.

1. If $m \vdash^{d,\alpha}_r \Gamma_0, C$ is an instance of (C-Ax) then $r = 0$ and $|C| = 1$. Hence C must be atomic. If C is false then the false atom lemma applies to give $m \vdash^{d,\alpha}_0 \Gamma_0$ and the result follows by weakening. Else we find C is true and thus $\neg C$ is false so apply the false atom lemma to the second assumption instead. This yields $m \vdash^{d',\alpha}_0 \Gamma_1$ and again the result follows by weakening.

2. All the remaining cases follow the pattern of reasoning given in the corresponding cut-rank reduction lemma for $EA_\infty(I;O)$. Hence we shall only highlight two particular cases.

i. Assume that the last rule applied to the derivation is universal quantification with C a side formula. Then we have the premises $\max(m,k) \vdash^{d_k,\alpha} \Gamma'_0, A(k), C$ for each $k \in \mathrm{N}$ where $d_k < d$. We apply weakening to the numerical declaration on the second assumption of the lemma to obtain $\max(m,k) \vdash^{d',\alpha} \Gamma_1, \neg C$. The induction hypothesis applied to these two sequents gives $\max(m,k) \vdash^{d'+d_k,\alpha}_r \Gamma'_0, A(k), \Gamma_1$. We may then re-apply universal quantification for the result as $d' + d_k < d' + d$.

ii. Now assume that the final rule of application is existential quantification with $C :\equiv \exists a C_0(a)$ the principal formulae. The premises of such a rule take the form

$$m \Vdash^{d_0,\alpha} k \qquad m \vdash^{d_1,\alpha}_r \Gamma_0, \exists a C_0(a), C_0(k)$$

where $d_0 < d_1 < d$. Applying the induction hypothesis to the right-hand premise with the second assumption of the lemma gives

$$m \vdash_r^{d'+d_1,\alpha} \Gamma_0, \Gamma_1, C_0(k). \tag{3.14}$$

Meanwhile we may use universal inversion and weakening on the second assumption to find

$$\max(m,k) \vdash_r^{d',\alpha} \Gamma_0, \Gamma_1, \neg C_0(k).$$

Using our bounding result and the computation of k in the left-hand premise of the $\exists$-rule we see that $m \Vdash^{d_0,\alpha} \max(m,k)$. Thus by (C-Cut)

$$m \vdash_r^{d'+d_1,\alpha} \Gamma_0, \Gamma_1, \neg C_0(k). \tag{3.15}$$

Since $|C_0(k)| \leq r$ we may apply (L-Cut) to (3.14) and (3.15) deducing as required

$$m \vdash_r^{d'+d,\alpha} \Gamma_0, \Gamma_1.$$

□

Theorem 3.18. ***Cut-Elimination.***

$$m \vdash_{r+1}^{d,\alpha} \Gamma \quad \Rightarrow \quad m \vdash_r^{2^d,\alpha} \Gamma$$

hence letting $d' := 2_r(d)$ *we find*

$$m \vdash_r^{d,\alpha} \Gamma \quad \Rightarrow \quad m \vdash_0^{d',\alpha} \Gamma.$$

Proof.

The first part uses induction over the height of the derivation. Note that there is no (C-Ax) case. The inductive steps are straightforward. We illustrate the most important case: an application of (L-Cut) on cut-formula with rank $r+1$. The premises will have the form

$$m \vdash_{r+1}^{d_0,\alpha} \Gamma, C \qquad m \vdash_{r+1}^{d_1,\alpha} \Gamma, \neg C$$

for $d_0, d_1 < d$ and $|C| = r + 1$. We apply the induction hypothesis to obtain

$$m \vdash_r^{2^{d_0},\alpha} \Gamma, C \qquad m \vdash_r^{2^{d_1},\alpha} \Gamma, \neg C.$$

For convenience we let $d' := \max(d_0, d_1)$ and match up the height of the sequents using weakening. We may now use the cut-rank reduction lemma (3.17) given above since one of the sequents will have the required form. We are left with $m \vdash_r^{2^{d'+1},\alpha} \Gamma$. As $d' < d$ implies $2^{d'+1} \leq 2^d$ we use weakening to leave the required form.

The full cut-elimination result now follows by a simple induction on r.

□

3.6 Embedding of EA^1

Theorem 3.19. ***Embedding of*** EA^1. *Assume that*

$$EA^1 \vdash \Gamma(a_0, \ldots, a_k)$$

where all the free variables are displayed.

Then this derivation determines some $\alpha \in E(\omega)$ and some $d, r \in \mathbf{N}$ such that, for all $m_0, \ldots, m_k$, if $m \geq \max(m_0, \ldots, m_k)$ then

$$m \vdash_r^{d,\alpha} \Gamma(m_0, \ldots, m_k).$$

Proof.

We use induction on the height of the EA^1 proof. The (L-Ax) case is straightforward. The inductive cases for disjunction, conjunction and cut follow easily from applying the induction hypothesis and re-applying the appropriate rule in EA^1_∞. Note that as there are two premises to the rules for conjunction and cut we will require weakening to change α'

and α'' to $\alpha' + \alpha''$ in the proof heights so that they match before re-applying the rule. We shall expand on three remaining cases.

1. (C-Ax). Let $\vec{a} := \vec{b}, \vec{c}, \vec{d}$ and assume $\Gamma(\vec{a}) :\equiv \Gamma'(\vec{c}), A(\vec{d}, \vec{b})$ where A is a Σ_1-formula and the final rule of inference is (C-Ax). Then we know $EA(I;O) \vdash A(\vec{x}; \vec{b})$. We may apply the embedding theorem of Chapter 2 on page 61 to obtain in $EA_\infty(I;O)$

$$n; m \vdash_r^{\omega \cdot h} A(n_0, \ldots, n_l, m_0, \ldots, m_j)$$

for some fixed $h, r \in \mathbf{N}$ where $n \geq \max(n_0, \ldots, n_l)$ and $m \geq \max(m_0, \ldots, m_j)$. Applying cut-elimination from 2.26 we obtain

$$n; m \vdash_0^{\alpha} A(n_0, \ldots, n_l, m_0, \ldots, m_j)$$

for $\alpha := 2_r(\omega \cdot h)$. Now we are in a position to use the (C-Ax) rule of EA^1_∞. It will give

$$\max(n, m) \vdash_0^{0,\alpha} A(n_0, \ldots, n_l, m_0, \ldots, m_j).$$

Finally, we use weakening for

$$m \vdash_0^{0,\alpha} \Gamma(m'_0, \ldots, m'_p, n_0, \ldots, n_l, m_0, \ldots, m_j)$$

where $m \geq \max(m'_0, \ldots, m'_p, n_0, \ldots, n_l, m_0, \ldots, m_j)$.

2. ($\forall$). The premise of the EA^1 derivation will be of the form $\Gamma', A(b)$ for b not free in Γ'. We apply the induction hypothesis for each possible assignment k to the free variable b and obtain

$$\max(m, k) \vdash_r^{d',\alpha} \Gamma', A(k)$$

for some fixed $\alpha \in E(\omega)$ and $d', r \in \mathbf{N}$. Immediately applying universal quantification in EA^1_∞ gives for $d := d' + 1$

$$m \vdash_r^{d,\alpha} \Gamma.$$

3. ($\exists$). In this case we would have a premise such as $\Gamma', A(t(\vec{b}))$. The variables $\vec{b}$ may include any number of the variables $\vec{a}$ in Γ as well as other extraneous variables not

amongst $\vec{a}$. We apply the induction hypothesis assigning 0 to these extraneous variables deducing

$$m \vdash_{r'}^{d_0,\alpha'} \Gamma', A(t(\vec{m})). \tag{3.16}$$

Now assume that $t(\vec{m})$ has value k. We may show from (L-Ax), by induction over the build-up of A,

$$m \vdash_{0}^{d_1,\alpha'} \neg A(t(\vec{m})), A(k) \tag{3.17}$$

for some finite d_1 dependent on the complexity of A. Then

$$\dfrac{\overset{\text{(L-Ax)}}{m \vdash_0^{0,\alpha'} t(\vec{m}) = k} \qquad \overset{\text{(3.17) and Weakening}}{m \vdash_0^{d_1,\alpha'} t(\vec{m}) \neq k, \neg A(t(\vec{m})), A(k)}}{m \vdash_1^{d_1+1,\alpha'} \neg A(t(\vec{m})), A(k).}\ \text{(L-Cut)}$$

Now by one further cut with (3.16), letting $r :=$ max$(r', |A|)$ and $d' := \max(d_0, d_1{+}1){+}1$

$$m \vdash_{r}^{d',\alpha'} \Gamma', A(k). \tag{3.18}$$

Now put $d := d' + 1$ and $\alpha := \omega \cdot (|t| + 1) + \alpha'$ where $|t|$ is the height of the term t.

$$\dfrac{\dfrac{\overset{\text{[Lemma 3.13]}}{m \Vdash^{0,\omega\cdot(|t|+1)} k}}{m \Vdash^{0,\alpha} k}\ \text{weakening} \qquad \dfrac{\overset{\text{(3.18)}}{m \vdash_r^{d',\alpha'} \Gamma', A(k)}}{m \vdash_r^{d',\alpha} \Gamma', A(k)}\ \text{weakening}}{m \vdash_r^{d,\alpha} \Gamma}\ (\exists)$$

□

3.7 EA^1 Is Σ_1 Conservative over $EA(I;O)$

Lemma 3.20. *Let Δ be a set of Σ_1-formulae. Assume that in EA^1_∞ we have a derivation*

$$m \vdash_0^{d,\alpha} \Delta.$$

Then if $|t|$ is maximum of the heights of any terms t in Δ and $\alpha' := \omega \cdot (|t| + 1) + \alpha$ we find Δ is true at $B_{d,\alpha'}(m)$.

Proof.

The proof follows in the same manner as the corresponding result for $EA_\infty(I;O)$ on page 65 using induction over the height of the derivation.

1. (C-Ax). If the derivation is a (C-Ax) then in $EA_\infty(I;O)$ we know $n_0, m_0 \vdash_0^\alpha \Delta$ for $\max(n_0, m_0) = m$. By Lemma 2.29 we find Δ is true at $B_{\alpha'}(n_0; m_0)$. The result follows by persistence since $B_{\alpha'}(n_0; m_0) \leq B_{\alpha'}(m; m) = B_{0,\alpha'}(m) \leq B_{d,\alpha'}(m)$.

2. (L-Ax). This case is trivially satisfied.

3. $(\vee)$ and $(\wedge)$. Given that $d' < d$ implies $B_{d',\alpha'}(m) \leq B_{d,\alpha'}(m)$ these cases follow just as in the proof of 2.29.

4. $(\forall)$. Since Δ is Σ_1 any universal quantifier must be a bounded quantifier. Let $\Delta :\equiv \Delta', \forall a(\neg a \leq t' \vee A(a))$. Then our premises are of the form

$$\{\text{max}(m, k) \vdash_0^{d_k,\alpha} \Delta', \neg k \leq t' \vee A(k)\}_{k \in \mathbf{N}}$$

where $d_k < d$. Inductively $\Delta', \neg k \leq t' \vee A(k)$ is true at $B_{d_k,\alpha'}(\text{max}(m, k))$ for every $k \in \mathbf{N}$.

If $\forall a(\neg a \leq t' \vee A(a))$ is true, then as a bounded formula it is automatically true at $B_{d,\alpha'}(m)$. Hence Δ is true at $B_{d,\alpha'}(m)$.

Else $\forall a(\neg a \leq t' \vee A(a))$ is false. Then by the induction hypothesis there exists some $k \leq t'$ such that Δ is true at $B_{d_k,\alpha'}(\text{max}(m, k))$. By 3.13 and 3.12 we find $k \leq B_{0,\omega\cdot(|t'|+1)}(m) \leq B_{d_k,\alpha'}(m)$. Hence by persistence Δ is true at

$$B_{d_k,\alpha'}(\text{max}(m, k)) \leq B_{d_k,\alpha'}(\text{max}(m, B_{d_k,\alpha'}(m)) \leq B_{d,\alpha'}(m).$$

5. $(\exists)$. Assume that $\Delta :\equiv \Delta', \exists a A(a)$ and we have premises

$$m \Vdash^{d_0,\alpha} k \qquad m \vdash_0^{d_1,\alpha} \Delta', A(k)$$

where $d_0 < d_1 < d$. Then inductively $\Delta', A(k)$ is true at $B_{d_1,\alpha'}(m)$. If Δ' is true, then by persistence it is true at $B_{d,\alpha'}(m)$. Hence Δ is true at $B_{d,\alpha'}(m)$. Else $A(k)$ is true at $B_{d_1,\alpha'}(m)$ which implies that Δ is true at $\max(k, B_{d_1,\alpha'}(m))$. From the computation of k in the premise of the rule we use weakening and the bounding lemma to find $k \leq B_{d_1,\alpha'}(m)$. Therefore Δ is true at $B_{d_1,\alpha'}(m) \leq B_{d,\alpha'}(m)$.

6. (C-Cut). If the last rule is a computational cut we have for $d_0, d_1 < d$ the premises

$$m \Vdash^{d_0,\alpha} k \qquad k \vdash_0^{d_1,\alpha} \Delta.$$

Let $d' := \max(d_0, d_1)$. We apply the induction hypothesis to find that Δ is true at $B_{d_1,\alpha'}(k) \leq B_{d',\alpha'}(k)$. The computation of k informs us that $k \leq B_{d_0,\alpha}(m) \leq B_{d',\alpha'}(m)$. Hence Δ is true at $B_{d',\alpha'}(B_{d',\alpha'}(m)) \leq B_{d,\alpha'}(m)$.

□

Theorem 3.21.

$$ProvRec(EA^1) \subseteq \mathcal{E}^3$$

Proof.

If the function f is provably recursive in EA^1 with bounded computational formula C_f then

$$EA^1 \vdash \forall\vec{a}\exists b\exists c(C_f(\vec{a}, b, c)).$$

Universal inversion gives

$$EA^1 \vdash \exists b\exists c(C_f(\vec{a}, b, c)).$$

By embedding this proof with 3.19 we obtain in EA^1_∞ for all assignments $\vec{a} := \vec{m}$ and some fixed $\alpha \in E(\omega)$, $d, r \in \mathbf{N}$

$$m \vdash_r^{d,\alpha} \exists b\exists c(C_f(\vec{m}, b, c))$$

where $m := \max(\vec{m})$. Then by cut-elimination 3.18 letting $d' := 2_r(d)$ we obtain

$$m \vdash_0^{d',\alpha} \exists b\exists c(C_f(\vec{m}, b, c)).$$

Now we may apply Lemma 3.20 above to find $\exists b \exists c(C_f(\vec{n}, b, c))$ is true at $B_{d',\alpha'}(m)$ where $\alpha' := \omega \cdot (|t| + 1) + \alpha$ for some $|t| \in \mathbf{N}$ determined by the original EA^1 derivation. As $B_{d',\alpha'}(m)$ is an elementary function in m by Corollary 3.9 the same argument applies as in the proof of the corresponding result from Chapter 2.

□

Corollary 3.22.

$$ProvRec(EA^1) = \mathcal{E}^3$$

Proof.

The previous result gives the left to right inclusion and Proposition 3.3 supplies the right to left inclusion.

□

Corollary 3.23. *EA^1 is conservative over $EA(I;O)$ for Σ_1-formulae with free variables.*

Proof.

Assume $EA^1 \vdash \exists \vec{d} C(\vec{a}, \vec{d})$ for a bounded formula C. Then by the results above there is an elementary function f such that for all $\vec{m}$ there exist $\vec{m'} \leq f(\vec{m})$ such that $C(\vec{m}, \vec{m'})$ is true.

Now define another elementary function f' which applied to $\vec{m}$ finds the least k such that $C(\vec{m}, (k)_0, \ldots, (k)_l)$ is true. Then define a computational formula for f' as

$$C_{f'}(\vec{m}, k) :\equiv C(\vec{m}, (k)_0, \ldots, (k)_l) \wedge \forall k' < k \neg C(\vec{m}, (k')_0, \ldots, (k')_l).$$

As f' is elementary $EA(I;O) \vdash \exists b C_{f'}(\vec{x}; b)$ and thus by logic $EA(I;O) \vdash \exists \vec{d} C(\vec{x}; \vec{d})$.

Note that the variables $\vec{a}$ from the EA^1 derivation have become input constants $\vec{x}$ which limits the conservativity result to Σ_1-formulae with free variables in accordance with the notion of provably recursive in $EA(I;O)$.

□

Chapter 4

EA^1(I;O) and *EA^2*

4.1 Introduction

We may now develop stronger arithmetic theories using the framework adopted in previous chapters.

Definition 4.1. ***The theory*** **$EA^1(I;O)$.**

$EA^1(I;O)$ is an extension of the theory EA^1 from Chapter 3. As the name suggests, $EA^1(I;O)$ renews the two-sorted variable separation considered in Chapter 1. We re-introduce the input constants, again denoted $x, y, z, \ldots$, which were eliminated by (C-Ax). Thus the language of $EA^1(I;O)$ is the same as $EA(I;O)$. The definitions of terms and atomic formulae in EA^1 are correspondingly extended to incorporate the new input constants. Naturally the quantification rules in EA^1 still only apply to the output variables $a, b, c, \ldots$ so that the inputs remain free constants. The only additional non-logical axiom we add to EA^1 in forming $EA^1(I;O)$ is the predicative induction axiom schema $\Gamma, A(0) \wedge \forall a(A(a) \rightarrow A(a+1)) \rightarrow A(x)$ for any formula A. Again we favour the rule formulation which is derivable from the axiom:

The **predicative induction** rule is, for an arbitrary set of formulae Γ

(P.Ind.) $$\frac{\Gamma, A(0) \qquad \Gamma, \neg A(a), A(a+1)}{\Gamma, A(x)}$$ where a is not free in Γ.

If $EA^1(I;O)$ did not have the closure axiom (C-Ax) then the theory would look identical to $EA(I;O)$. However by defining EA^1 as a stepping stone to $EA^1(I;O)$ we have layered and thus restricted the use of (C-Ax). If we had attempted to define $EA^1(I;O)$ as $EA(I;O)+$ (C-Ax) without careful stratification we would obtain a much stronger theory. This point is demonstrated in the following chapter.

All of the derivations given in Chapter 1 may be carried out in $EA^1(I;O)$. We adopt the two-sorted definition for provably recursive functions given in 1.15 on page 25 for $EA^1(I;O)$. Clearly any function provably recursive in EA^1 will also be provably recursive in $EA^1(I;O)$ once we invert the universally quantified $\vec{a}$ in Definition 3.2 at inputs $\vec{x}$. Hence $ProvRec(EA^1) \subseteq ProvRec(EA^1(I;O))$.

Definition 4.2. ***The theory*** EA^2**.**

EA^2 is now defined as a theory of closure over $EA^1(I;O)$ following exactly the approach used in Chapter 3 to define EA^1 from $EA(I;O)$. Hence EA^2 is a new theory with just one type of variable, output variables, which has the usual logical axioms and rules for disjunction, conjunction, quantification and cut. It has all the non-logical axioms for equality, basic arithmetic and coding but no predicative induction. EA^2 includes a closure axiom which now incorporates derivations from $EA^1(I;O)$.

The Σ_1**-closure axiom** of EA^2 reads,

(C-Ax) $$EA^2 \vdash \Gamma(\vec{c}), A(\vec{a}, \vec{b}) \qquad \text{if} \quad EA^1(I;O) \vdash A(\vec{x}; \vec{b}).$$

where A is a Σ_1-formula, Γ is an arbitrary set of formulae, and where all the free variables of Γ, A are indicated.

The numerical super-script in EA^2 indicates the maximum possible number of nested applications of (C-Ax) used in any derivation. Since EA^2 has only one type of variable we use the single-sorted definition for a function to be provably recursive in the same way as we did for EA^1 in 3.2 on page 72. Using 3.3 we immediately conclude $ProvRec(EA^1(I;O)) \subseteq ProvRec(EA^2)$.

4.2 Lower Bounds for Provably Recursive Functions

In this section we give lower bounds on the provably recursive functions of both $EA^1(I;O)$ and EA^2. Then in the following two sections we provide corresponding upper bounds by developing appropriate infinitary theories.

Theorem 4.3. *$ProvRec(EA^1(I;O))$ is closed under a single primitive recursion from functions in $ProvRec(EA^1)$.*

That is, for $k > 0$ if the k-ary function g and the $k+2$-ary function h are provably recursive in EA^1 then for inputs $\vec{y} := y_1, \ldots, y_k$ and x the function f defined by the primitive recursion

$$f(\vec{y}, 0) = g(\vec{y}) \qquad f(\vec{y}, x+1) = h(\vec{y}, x, f(\vec{y}, x))$$

is provably recursive in $EA^1(I;O)$.

Proof.

For clarity, and without loss of generality, let k be 1. Assume that we are given computational formulae C_g and C_h for the functions g and h respectively. We look to define a computational formula C_f for the function f. We shall compute $f(e, a) = b$ (where e, a are output variables which will later be replaced by inputs y_1, x to adhere to the definition of a provably recursive function in $EA^1(I;O)$). We let c be the code for a sequence $\langle c_0, \ldots, c_a \rangle$. In this sequence each component c_d codes a pair whose left inverse gives the value of $f(e,d)$ and whose right inverse gives the code for the computation of $f(e,d)$. Formally our definition of C_f reads

$$\begin{aligned} C_f(e,a,b,c) \quad :\equiv \quad & lh(c) = a+1 \wedge \forall d < a+1((c)_d \neq 0) \\ & \wedge l((c)_a) = b \wedge C_g(e, l((c)_0), r((c)_0)) \\ & \wedge \forall d < a(C_h(e,d,l((c)_d), l((c)_{d+1}), r((c)_{d+1}))). \end{aligned} \tag{4.1}$$

Working in $EA^1(I;O)$ we prove the existence condition for f follows from showing $\exists b \exists c C_f(e,a,b,c)$ is progressive in a. We argue in a natural deduction style with extensive use of the coding axioms on page 14.

i. Assume that we are provided a b and a c such that $C_g(e,b,c)$. Then let $t = p(0, p(b,c))$. We now deduce $C_f(e, 0, l((t)_0), t)$ working through the five conjuncts in the definition above.

Firstly $l(t) = 0$ so $lh(t) = lh(l(t)) + 1 = lh(0) + 1 = 0 + 1$. Secondly $(t)_0 = r(l^0(t)) = p(b,c) \neq 0$ thus $\forall d < 0 + 1((t)_d \neq 0)$. The third conjunct is an equality axiom. The fourth follows from $C_g(e,b,c)$ as $l((t)_0) = b$ and $r((t)_0) = c$. Finally $\neg d < 0$ holds thus the last conjunct follows by weakening. Bringing in our assumption and quantifying we have

$$EA^1(I;O) \vdash \neg \forall e \exists b \exists c C_g(e,b,c), \exists b \exists c C_f(e,0,b,c). \tag{4.2}$$

Since g is provably recursive in EA^1 we have a derivation of $\forall e \exists b \exists c C_g(e,b,c)$ in $EA^1(I;O)$. A cut with (4.2) leaves

$$EA^1(I;O) \vdash \exists b \exists c C_f(e,0,b,c). \tag{4.3}$$

ii. Now assume that we have a b and c such that $C_f(e,a,b,c)$ holds and a b' and c' such that $C_h(e,a,b,b',c')$. Let t be the term $p(c, p(b',c'))$ in order to deduce $C_f(e, a+1, b', t)$. Again there are five conjuncts to check.

Firstly by our assumption for b and c we have $lh(c) = a+1$. Thus $lh(t) = lh(l(t))+1 = lh(c)+1 = a+2$. Secondly if $d < a+2$ then either $d < a+1$ or $d = a+1$. In the former case by the assumption for c we have $(c)_d \neq 0$. Using Lemma 1.17 we deduce $(t)_d \neq 0$. In the latter case $(t)_{a+1} = r(l^0(t)) = r(t) = p(b', c') \neq 0$. The third conjunct follows from $l((t)_{a+1}) = l(p(b', c')) = b'$. The fourth conjunct follows from the assumption $C_f(e, a, b, c)$ since it contains $C_g(e, l((c)_0), r((c)_0))$ and $(c)_0 = (t)_0$ by Lemma 1.17.

The last conjunct has two cases. Assuming $d < a+1$ either $d < a$ or $d = a$. If $d < a$ then by the assumption $C_f(e, a, b, c)$ we have

$$C_h(e, d, l((c)_d), l((c)_{d+1}), r((c)_{d+1})).$$

Following Lemma 1.17 $(c)_d = (t)_d$ and $(c)_{d+1} = (t)_{d+1}$ and thus

$$C_h(e, d, l((t)_d), l((t)_{d+1}), r((t)_{d+1})).$$

On the other hand if $d = a$ then the conjunct follows from the assumption $C_h(e, a, b, b', c')$ since $l((t)_a) = l((c)_a) = b$, $l((t)_{a+1}) = b'$ and $r((t)_{a+1}) = c'$.

Collating our assumptions yields

$$EA^1(I;O) \vdash \neg C_f(e, a, b, c), \neg C_h(e, a, b, b', c'), C_f(e, a+1, b', t)$$

from which applying quantifiers in the correct order leaves

$$EA^1(I;O) \vdash \neg\exists b\exists c C_f(e, a, b, c), \neg\forall e\forall a\forall b\exists b'\exists c C_h(e, a, b, b', c), \exists b\exists c C_f(e, a+1, b, c). \tag{4.4}$$

As h is provably recursive in EA^1 we have a derivation of $\forall e\forall a\forall b\exists b'\exists c C_h(e, a, b, b', c)$. A cut with (4.4) gives

$$EA^1(I;O) \vdash \neg\exists b\exists c C_f(e, a, b, c), \exists b\exists c C_f(e, a+1, b, c). \tag{4.5}$$

We may now replace the output e by an input y_1 and apply predicative induction to the derivations (4.3) and (4.5) for

$$EA^1(I;O) \vdash \exists b\exists c C_f(y_1, x; b, c).$$

This gives the existence condition for f to be provably recursive in $EA^1(I;O)$. The uniqueness condition for $f(e,a)$ can be easily shown to be progressive in a using the definition of $C_f(e,a,b,c)$ and the coding axioms. The base case will follow from the uniqueness condition for g whilst the inductive step makes use of uniqueness for h.

$\square$

Corollary 4.4. *The super-exponential function,* $s(n)$*, can be defined by the primitive recursion* $s(0) = 1$ *and* $s(n+1) = 2^{s(n)}$*. As the exponential function is provably recursive in* EA^1 *we find from the previous theorem that* $s(x)$ *is provably recursive in* $EA^1(I;O)$*.*

The super-exponential function can be seen as a 'backbone' of Grzegorczyk's class $\mathcal{E}^4$. That is, for any $f \in \mathcal{E}^4$ and a fixed $k \in \mathbf{N}$ we have $f(\vec{n}) \leq s^k(\max(\vec{n}))$ where $s^0(n) = n$ and $s^{k+1}(n) = s(s^k(n))$ (cf. [43] or [51]). However in attempting to show that these finite iterates are provably recursive in $EA^1(I;O)$ we again face the problem of input substitution in two-sorted theories. In $EA(I;O)$ this issue did not prevent closure under composition of $ProvRec(EA(I;O))$ as $ProvRec(EA(I;O)) = \mathcal{E}^3$ and $\mathcal{E}^3$ *is* closed under composition. Now in $EA^1(I;O)$, without input substitution we quickly reach a barrier: $ProvRec(EA^1(I;O))$ is not closed under composition. For example, we prove in the next section of this chapter that $s(s(x))$ is not provably recursive. What we can do is apply the Gentzen argument used in Lemma 1.23 on page 34 for $EA(I;O)$ to show the input x in the super-exponential function may be replaced by the exponential stack $2_k(x)$ for a fixed $k \in \mathbf{N}$.

Fact 4.5.

Recall that for a fixed $k \in \mathbf{N}$ we denote the k-times iterate of 2^x as $2_k(x)$. We let $\mathcal{E}^4_1$ denote the class of functions computable in a number of steps bounded by $s(2_k(t(\vec{n})))$ for any fixed $k \in \mathbf{N}$ and any polynomial t. This class forms the first level of a proper sub-hierarchy $\mathcal{E}^4_i$ such that $\bigcup_{i \in \mathbf{N}} \mathcal{E}^4_i = \mathcal{E}^4$. We refer the reader to [11], [27] or [28] for verification.

Theorem 4.6.

$$ProvRec(EA^1(I;O)) \supseteq \mathcal{E}_1^4.$$

Proof.

Assume $f \in \mathcal{E}_1^4$. Then we may assume there is a program P for an unlimited register machine M which computes $f(\vec{x})$ in a number of steps bounded by $s(2_k(t(\vec{x})))$ for a fixed $k \in \mathbf{N}$ and polynomial t. Recall the proof of Theorem 1.24 in Chapter 1. We showed that there is suitable bounded computational formula $C_M(\vec{x};d,b,c)$ such that c codes d-many steps of the computation by M of $f(\vec{x}) = b$. Moreover, we proved $\exists b\exists c C_M(\vec{x};d,b,c)$ is progressive in d. Working in $EA(I;O)$ we may apply Lemma 1.23 with $k := 1$ and $t := y$ to deduce

$$EA(I;O) \vdash \exists b\exists c C_M(\vec{x}, 2^y; b, c).$$

This is, by Definition 1.21,

$$EA(I;O) \vdash \exists b_0(\exists c_0 E(y;b_0,c_0) \wedge \exists b\exists c C_M(\vec{x};b_0,b,c)).$$

By logic this formula is provably equivalent to a Σ_1-formula. Moving to the theory $EA^1(I;O)$, using (C-Ax) and then universal quantification we may deduce

$$EA^1(I;O) \vdash \forall\vec{a}\forall e\exists b_0(\exists c_0 E(e,b_0,c') \wedge \exists b\exists c C_M(\vec{a},b_0,b,c)). \tag{4.6}$$

Now let $2^e = b_0$ be shorthand for $\exists c' E(e,b_0,c')$. Likewise put $s(d) = b_1 :\equiv \exists c S(d,b_1,c)$ where $\exists c S(d,b_1,c)$ is a computational formula defining the super-exponential function using a sequence code c. S may be formally defined by following the proof of Theorem 4.3. The same theorem will show

$$EA^1(I;O) \vdash s(0) = 1 \quad \text{and} \quad EA^1(I;O) \vdash s(d) = b_1 \wedge 2^{b_1} = b_0 \rightarrow s(d+1) = b_0. \tag{4.7}$$

We now claim

$$EA^1(I;O) \vdash Prog_d \exists b_1(s(d) = b_1 \wedge \exists b\exists c C_M(\vec{x};b_1,b,c)). \tag{4.8}$$

i. We know $Prog_d \exists b \exists c C_M(\vec{x}; d, b, c)$ from 1.24. Hence $\exists b \exists c C_M(\vec{x}; 1, b, c)$. Since $s(0) = 1$ is also provable, (4.7), the base case follows by conjunction and existential quantification at 1.

ii. Assume we are given a b_1 such that $s(d) = b_1$ and $\exists b \exists c C_M(\vec{x}; b_1, b, c)$. For this b_1 we invert the $\forall e$ in (4.6) at b_1 and invert the $\forall \vec{a}$ at $\vec{x}$. We therefore find there is a b_0 such that $2^{b_1} = b_0$ and $\exists b \exists c C_M(\vec{x}; b_0, b, c)$. Since $s(d) = b_1$ and $2^{b_1} = b_0$ we deduce $s(d+1) = b_0$ from (4.7). Thus by conjunction and existential quantification with witness b_0 we obtain

$$\exists b_1 (s(d+1) = b_1 \wedge \exists b \exists c C_M(\vec{x}; b_1, b, c)).$$

This gives the inductive step for (4.8).

Using (4.8) we may apply Lemma 1.23 from Chapter 1 to conclude

$$EA^1(I;O) \vdash \exists b_1 (s(2_k(t(\vec{x}))) = b_1 \wedge \exists b \exists c C_M(\vec{x}; b_1, b, c)).$$

As $f(\vec{x})$ is computable by P in a number of steps bounded by $2_k(t(\vec{x}))$, this sequent (which is provably equivalent to a Σ_1-formula) gives the existence condition for f to be provably recursive in $EA^1(I;O)$. The uniqueness condition follows by applying the same argument.

□

In order to capture all of the Grzegorczyk's class $\mathcal{E}^4$ we must move to those functions provably recursive in EA^2. We shall use a characterization of the Grzegorczyk Hierarchy above $\mathcal{E}^3$ proved by Axt in [3].

Fact 4.7. *(Axt [3]) For* $i \geq 3$*, the Grzegorczyk class* $\mathcal{E}^{i+1}$ *is the smallest class of functions containing* $\mathcal{E}^i$ *which is closed under composition and closed under a single primitive recursion.*

Theorem 4.8.

$$ProvRec(EA^2) \supseteq \mathcal{E}^4.$$

Proof.

Assume $f \in \mathcal{E}^4$. We use induction over the definition of f following Fact 4.7 above.

1. Firstly it may be that $f \in \mathcal{E}^3$. Then f is provably recursive in EA^2 since by 3.22 $\mathcal{E}^3 = ProvRec(EA^1)$ and $ProvRec(EA^1) \subset ProvRec(EA^2)$.

2. Assume that f is defined by composition where the auxiliary functions are all members of $\mathcal{E}^4$ and thus, by the induction hypothesis, already provably recursive in EA^2. Then as Theorem 3.4 applies equally well to EA^2 we know $ProvRec(EA^2)$ is closed under composition. Hence f is provably recursive in EA^2.

3. Finally f may be defined by a primitive recursion. The auxiliary functions are all elementary and thus provably recursive in EA^1 by 3.22. But then following Theorem 4.3 we see f is provably recursive in $EA^1(I;O)$ and thus it is also provably recursive in EA^2.

□

4.3 Upper Bounds for Provably Recursive Functions

In this section we shall prove the converse inclusions for Theorem 4.6 and Theorem 4.8 above. Following the methods in Chapters 2 and 3 we find upper bounds on the provably recursive functions of $EA^1(I;O)$ and EA^2 by developing appropriate infinitary theories. *Aside from the definitions of the appropriate bounding functions and the proofs of their required properties, the proofs of structural rules, cut-elimination results and embeddings are all essentially the same as the corresponding results in Chapters 2 and 3.*

4.3.1 An Infinitary Theory for $EA^1(I;O)$

Definition 4.9. ***The infinitary theory*** $EA^1_\infty(I;O)$.

$EA^1_\infty(I;O)$ extends the theory EA^1_∞ from Chapter 3. We shall now re-introduce the assumption of an input domain of numbers alongside the output domain. Hence to the left of the proof gate, as in $EA_\infty(I;O)$, we have two natural number 'declarations'. The finite measure d from EA^1_∞ now becomes a structured tree-ordinal α_1 whose assignment in rules of inference is controlled by the input parameter n. This allows us to embed inductions in $EA^1(I;O)$ into the infinitary theory. Hence proof height is now considered as the 'composition' of two ordinals, $\alpha_1 \circ \alpha_0$ or 'α_0 then α_1'. Otherwise we have precisely the same set of rules as EA^1_∞. A sequent in $EA^1_\infty(I;O)$ takes the form

$$n; m \vdash^{\alpha_1, \alpha_0}_{r} \Gamma.$$

This is intended to be read as 'given fixed natural number parameters $\leq n$ from the input domain and values $\leq m$ from the output domain, the truth of Γ (in the standard model) can be established in ($\alpha_1 \circ \alpha_0$)-many steps using cuts on formulae whose rank is $\leq r$'. These sequents are defined inductively in the following.

Logical Rules

(C-Ax)	$n; \max(n_0, m_0) \vdash_0^{\alpha_1,\alpha_0} \Gamma$	for any α_1, n, if in $EA_\infty(I;O)$ we already have $n_0; m_0 \vdash_0^{\alpha_0} \Gamma$.
(L-Ax)	$n; m \vdash_r^{\alpha_1,\alpha_0} \Gamma$	for any α_1, α_0 and r, if Γ contains a true atom.
$(\vee)$	$\dfrac{n; m \vdash_{r_i}^{\beta_i,\alpha_0} \Gamma, A_i}{n; m \vdash_r^{\alpha_1,\alpha_0} \Gamma, A_0 \vee A_1}$	if $\beta_i \in \alpha_1[n]$ and $r_i \leq r$ where $i = 0$ or 1.
$(\wedge)$	$\dfrac{n; m \vdash_{r_0}^{\beta_0,\alpha_0} \Gamma, A_0 \qquad n; m \vdash_{r_1}^{\beta_1,\alpha_0} \Gamma, A_1}{n; m \vdash_r^{\alpha_1,\alpha_0} \Gamma, A_0 \wedge A_1}$	if $\beta_0, \beta_1 \in \alpha_1[n]$ and $r_0, r_1 \leq r$.
$(\forall)$	$\dfrac{\{n; \max(m, k) \vdash_{r'}^{\beta_k,\alpha_0} \Gamma, A(k)\}_{k \in \mathbb{N}}}{n; m \vdash_r^{\alpha_1,\alpha_0} \Gamma, \forall a A(a)}$	if for all $k \in \mathbf{N}$, $\beta_k \in \alpha_1[n]$ and $r' \leq r$.
$(\exists)$	$\dfrac{n; m \Vdash^{\beta_0,\alpha_0} k \qquad n; m \vdash_{r_1}^{\beta_1,\alpha_0} \Gamma, A(k)}{n; m \vdash_r^{\alpha_1,\alpha_0} \Gamma, \exists a A(a)}$	if $\beta_0 \in \beta_1[n]$, $\beta_1 \in \alpha_1[n]$ and $r_1 \leq r$.
(L-Cut)	$\dfrac{n; m \vdash_{r_0}^{\beta_0,\alpha_0} \Gamma, \neg C \qquad n; m \vdash_{r_1}^{\beta_1,\alpha_0} \Gamma, C}{n; m \vdash_r^{\alpha_1,\alpha_0} \Gamma}$	if $\beta_0, \beta_1 \in \alpha_1[n]$ and $\lvert C \rvert, r_0, r_1 \leq r$.
(C-Cut)	$\dfrac{n; m \Vdash^{\beta_0,\alpha_0} k \qquad n; k \vdash_{r_1}^{\beta_1,\alpha_0} \Gamma}{n; m \vdash_r^{\alpha_1,\alpha_0} \Gamma}$	if $\beta_0, \beta_1 \in \alpha_1[n]$ and $r_1 \leq r$.

Computational Rules

(𝕆-Ax) $\quad n; \max(n_0, m_0) \Vdash^{\alpha_1,\alpha_0} k \quad$ for any α_1 and n, if $k \le B_{\alpha_0}(n_0; m_0)$.

(𝕆-Cut) $$\frac{n; m \Vdash^{\beta_0,\alpha_0} m' \qquad n; m' \Vdash^{\beta_1,\alpha_0} k}{n; m \Vdash^{\alpha_1,\alpha_0} k}$$ if $\beta_0, \beta_1 \in \alpha_1[n]$.

The relationship between $EA^1_\infty(I;O)$ and EA^1_∞ may be illuminated by the following result.

Lemma 4.10. *If in* $EA^1_\infty(I;O)$ *we have* $n; m \vdash_r^{\alpha_1,\alpha_0} \Gamma$ *then in* EA^1_∞ *we find* $m \vdash_r^{d,\alpha_0} \Gamma$ *where* $d := G_{\alpha_1}(n)$.

Proof.

We use induction over the derivation of Γ in $EA^1_\infty(I;O)$ with a sub-induction proving the same correspondence for the computational sequents. The argument is straightforward since in the axioms α_1 is arbitrary and for any rule of inference with $\beta \in \alpha_1[n]$ we deduce $G_\beta(n) < G_{\alpha_1}(n)$ and may re-apply the same rule in EA^1_∞.

□

Note similarly to Lemma 2.9 that the converse is not necessarily true. For example $n+1 < G_{\omega+1}(n)$ but $n+1 \notin \omega+1[n]$.

Recall the definition of the bounding function $B_{d,\alpha}(m)$ used for EA^1_∞ (3.8). To give a bounding function for $EA^1(I;O)$ we simply include an additional numerical argument n and extend the inductive definition to include a limit stage by diagonalising over n.

Definition 4.11. *For* $\alpha_1, \alpha_0 \in \Omega^S$ *the functions* $B_{\alpha_1,\alpha_0} : \mathbf{N}^2 \to \mathbf{N}$ *are defined by recursion over* α_1*:*

$$\begin{aligned} B_{0,\alpha_0}(n;m) &:= B_{0,\alpha_0}(m), \\ B_{\alpha_1+1,\alpha_0}(n;m) &:= B_{\alpha_1,\alpha_0}(n; B_{\alpha_1,\alpha_0}(n;m)), \\ B_{\lambda,\alpha_0}(n;m) &:= B_{\lambda_n,\alpha_0}(n;m). \end{aligned}$$

Note from the definition that $B_{0,\alpha_0}(n;m)$ is defined in terms of a function on α_0 and m only.

Corollary 4.12.

$$B_{\alpha_1,\alpha_0}(n;m) = B_{d,\alpha_0}(m) \quad \textit{where } d := G_{\alpha_1}(n).$$

Hence given $\alpha_1, \alpha_0 \in E(\omega)$ the function $B_{\alpha_1,\alpha_0} : \mathbf{N}^2 \to \mathbf{N}$ is in Grzegorczyk's class $\mathcal{E}^4$.

Proof.

The first claim follows by a simple induction over α_1. Using the standard notion for iterates of a function it is easily seen from the definition of $B_{d,\alpha_0}(m)$ that we may also write $B_{\alpha_1,\alpha_0}(n;m) = B^k_{0,\alpha_0}(m)$ where $k := 2^{G_{\alpha_1}(n)}$.

Recall that for any $\alpha_0 \in E(\omega)$ the function $B_{0,\alpha_0}(m) = B_{\alpha_0}(m;m)$ is an elementary function in m, (2.12). Then, as a function of d and m, we find the iterate $B_{d,\alpha_0}(m)$ is in $\mathcal{E}^4$ (cf. [51]). From the first claim it follows that for any $\alpha_1 \in E(\omega)$, the function $B_{\alpha_1,\alpha_0}(n;m)$ also belongs to $\mathcal{E}^4$ by composition.

□

We may in fact refine the position of B_{α_1,α_0} in Grzegorczyk's class $\mathcal{E}^4$. Recall we denote by $\mathcal{E}^4_1$ the class of functions computable in a number of steps bounded by $s(2_k(t(\vec{n})))$ for some fixed $k \in \mathbf{N}$ and polynomial t. We shall make use of the two place super-exponential function $s(n,m)$ defined as $s(0,m) := m$ and $s(n+1,m) := 2^{s(n,m)}$. This function is often written with the notation $2_n(m)$ but we shall avoid having arguments in sub-scripts for clarity. A simple induction shows $s(n,m) \leq s(n+m)$.

Lemma 4.13. *For any $\alpha_1, \alpha_0 \in E(\omega)$ the function $B_{\alpha_1,\alpha_0} : \mathbf{N}^2 \to \mathbf{N}$ is in $\mathcal{E}^4_1$.*

Proof.

By the comments above it will suffice to show that for a given $\alpha_1, \alpha_0 \in E(\omega)$, the number of steps required to compute $B_{\alpha_1,\alpha_0}(n;m)$ is bounded by $s(2_k(n), m)$ for some fixed $k \in \mathbf{N}$.

As $B_{0,\alpha_0}(m)$ is elementary there exists some fixed $l \in \mathbf{N}$ such that both the value of the function and the number of steps required to compute the function are bounded by $2_l(m)$. By the the preceding corollary we know $B_{\alpha_1,\alpha_0}(n;m)$ is given by iterating $B_{0,\alpha_0}(m)$ r-many times where $r := 2^{G_{\alpha_1}(n)}$ is another elementary function (in n).

Thus a computation of $B_{\alpha_1,\alpha_0}(n;m)$ requires a number of steps given by

$$\begin{aligned} & 2_l(m) \\ + & \, 2_l(2_l(m)) \\ + & \, 2_l\big(2_l(m) + 2_l(2_l(m))\big) \\ + & \, \ldots \end{aligned}$$

where there are r-many terms in the sum. We may write this sum as $\sum_{i=1}^{r} a_i$ where $a_1 = 2_l(m)$ and for $1 < j \leq r$ we have $a_j = 2_l\big(\sum_{i=1}^{j-1} a_i\big)$.

We prove by induction on j that $\sum_{i=1}^{j} a_i \leq 2_{j(l+1)}(m)$. The base case is trivial. For the inductive step, noting that $2_l(m) + 2_l(m) \leq 2_{l+1}(m)$, we have

$$\sum_{i=1}^{j+1} a_i \leq 2_{j(l+1)}(m) + 2_l(2_{j(l+1)}(m)) \leq 2_{j(l+1)+l}(m) + 2_{j(l+1)+l}(m) \leq 2_{(j+1)(l+1)}(m).$$

Therefore the number of steps to compute $B_{\alpha_1,\alpha_0}(n;m)$ is bounded by $2_{r(l+1)}(m)$ which may also be written as $s(r(l+1), m)$. Since $r := 2^{G_{\alpha_1}(n)}$ we know $r \leq 2_{k'}(n)$ for a fixed $k' \in \mathbf{N}$. A straightforward induction gives $2_{k'}(n) \cdot (l+1) \leq 2_{k'+l}(n)$ and thus $B_{\alpha_1,\alpha_0}(n;m)$ is computed in a number of steps bounded by $s(2_k(n), m)$ where $k := k' + l$.

$\square$

Lemma 4.14. *For $\alpha_1, \gamma_1 \in \Omega^S$ and $m, n \in \mathbf{N}$:*

1. If $\beta \in \alpha_1[n]$ then $B_{\beta,\alpha_0}(n;m) < B_{\beta+1,\alpha_0}(n;m) \leq B_{\alpha_1,\alpha_0}(n;m)$.

2. $B_{\alpha_1,\alpha_0}(n;m)$ is strictly increasing in m and increasing in n, strictly so if α_1 is infinite.

3. $B_{\alpha_1,\alpha_0}(n;m) \leq B_{\beta_1+\alpha_1,\alpha_0}(n;m)$.

Proof.

Since $B_{\alpha_1,\alpha_0}(n;m)$ is just $B_{d,\alpha_0}(n;m)$ for $d := G_{\alpha_1}(n)$ these results follow immediately from Lemma 2.13 and Lemma 3.10.

$\square$

Lemma 4.15. ***Bounding for*** $EA^1_\infty(I;O)$**.**

$$n;m \Vdash^{\alpha_1,\alpha_0} k \qquad \Leftrightarrow \qquad k \leq B_{\alpha_1,\alpha_0}(n;m).$$

Proof.

For the left to right implication we know $n;m \Vdash^{\alpha_1,\alpha_0} k$ implies that in EA^1_∞ we have a computation $m \Vdash^{d,\alpha_0} k$ for $d := G_{\alpha_1}(n)$ from the method of Lemma 4.10. Hence we simply apply the bounding result for EA^1_∞, Lemma 3.11, and then use Corollary 4.12. The right to left implication uses induction on α_1 in exactly the same manner as the bounding result for $EA_\infty(I;O)$ given in Lemma 2.14.

$\square$

Lemma 4.16. ***Weakening for Computations.*** *If we have the computation*

$$n;m \Vdash^{\alpha_1,\alpha_0} k$$

and if $n \leq n', m \leq m', k \geq k'$ *and* $\alpha_1[n] \subseteq \alpha_1'[n]$, $\alpha_0[m] \subseteq \alpha_0'[m]$ *then for any* $\gamma, \delta \in \Omega^S$ *then we also have the computation*

$$n';m' \Vdash^{\gamma+\alpha_1',\delta+\alpha_0'} k'.$$

Proof.

We use induction over the height of the computation. If the computation is an instance of ($\mathbb{O}$-Ax) then $k \leq B_{\alpha_0}(n_0;m_0)$ where $\max(n_0,m_0) = m$. Thus by the corresponding result in Chapter 2 we find $k' \leq B_{\delta+\alpha_0'}(m';m')$. Since either $\alpha_1' = 0$ or $0 \in \alpha_1'[n]$ it follows from 4.14 above that $k' \leq B_{\gamma+\alpha_1',\delta+\alpha_0'}(n';m')$. Hence the result follows by the previous lemma. Now consider the case where the last rule applied is ($\mathbb{O}$-Cut). Then after applying the induction hypothesis, since $\beta \in \alpha_1[n] \Rightarrow \gamma + \beta \in \gamma + \alpha_1'[n']$ we re-apply ($\mathbb{O}$-Cut) to give the required result.

□

Lemma 4.17. *For any term* $t(\vec{x};\vec{a})$ *in* $EA^1(I;O)$ *and any numbers* m *and* n*, if the value of the term upon substituting* $m_i \leq m$ *for each* a_i *and* $n_i \leq n$ *for each* x_i *is the number* k *we have in* $EA^1_\infty(I;O)$ *the computation*

$$n;m \Vdash^{\omega\cdot(|t|+1),0} k.$$

Proof.

We show $B_{\alpha_1,0}(n;m) = B_{\alpha_1}(n;m)$ whence the result will follow by Lemma 2.17 and Lemma 2.14 in Chapter 2 and Lemma 4.15 above.

When $\alpha_1 = 0$ we find $B_{0,0}(n;m) = B_0(m) = B_0(m;m) = p(m) = B_0(n;m)$. Now assume α_1 is a successor. Then using the induction hypothesis

$$B_{\alpha_1+1,0}(n;m) = B_{\alpha_1,0}(n;B_{\alpha_1,0}(n;m)) = B_{\alpha_1}(n;B_{\alpha_1}(n;m)) = B_{\alpha_1+1}(n;m).$$

Similarly when α_1 is a limit, $B_{\lambda,0}(n;m) = B_{\lambda_n,0}(n;m) = B_{\lambda_n}(n;m) = B_\lambda(n;m)$.

□

We now derive the usual structural rules for $EA^1_\infty(I;O)$ in order to prove cut-elimination.

Lemma 4.18. ***Weakening for Logical Rules.*** *If*

$$n;m \vdash^{\alpha_1,\alpha_0} \Gamma$$

and if $n \leq n', m \leq m', \Gamma \subseteq \Gamma'$ *and* $\alpha_1[n] \subseteq \alpha_1'[n]$, $\alpha_0[m] \subseteq \alpha_0'[m]$ *then for any* $\gamma, \delta \in \Omega^S$

$$n';m' \vdash^{\gamma+\alpha_1',\delta+\alpha_0'} \Gamma'.$$

Lemma 4.19. ***Inversions.***

$$n;m \vdash_r^{\alpha_1,\alpha_0} \Gamma, A_0 \wedge A_1 \quad \Rightarrow \quad n;m \vdash_r^{\alpha_1,\alpha_0} \Gamma, A_i \qquad \textit{where } i = 0 \textit{ or } 1.$$

$$n;m \vdash_r^{\alpha_1,\alpha_0} \Gamma, \forall a A(a) \quad \Rightarrow \quad n;\max(m,k) \vdash_r^{\alpha_1,\alpha_0} \Gamma, A(k) \quad \textit{for every } k \in \mathbf{N}.$$

Proofs.

In each case we use induction over the height of the derivation. Where the derivation is an instance (C-Ax) we use the argument presented in the proofs of the corresponding Lemma 3.14 and Lemma 3.15 for EA^1_∞ in Chapter 3. Then in each of the other cases the reasoning matches that used in Chapter 2 for $EA_\infty(I;O)$. That is, letting α be α_1 and suppressing the mention of α_0 we follow exactly the proofs of Lemma 2.18, Lemma 2.19 and Lemma 2.20.

□

Theorem 4.20. ***Cut-Elimination.*** *Letting* $\delta_1 := 2_r(\alpha_1)$*,*

$$n;m \vdash_r^{\alpha_1,\alpha_0} \Gamma \quad \Rightarrow \quad n;m \vdash_0^{\delta_1,\alpha_0} \Gamma$$

Proof.

Firstly we need to prove a false-atom lemma and a cut-rank reduction lemma. As above, we use induction of the height of the derivation with the corresponding results for EA^1_∞ in Chapter 3 sufficing for cases of (C-Ax) and the corresponding results for $EA_\infty(I;O)$ in Chapter 2 used in all other cases. Then cut-reduction and cut-elimination follow using the proofs for $EA_\infty(I;O)$ in Chapter 2.

□

Theorem 4.21. ***Embedding of*** $EA^1(I;O)$. *Assume*

$$EA^1(I;O) \vdash \Gamma(x_0,\dots,x_l;a_0,\dots,a_k)$$

where all the free variables are displayed.

Then this derivation determines some $\alpha_0 \in E(\omega)$ *and some* $d,r \in \mathbf{N}$ *such that, for all* $n_0,\dots,n_l$ *and all* $m_0,\dots,m_k$, *if* $n \geq \max(n_0,\dots,n_l)$ *and* $m \geq \max(m_0,\dots,m_k)$ *then*

$$n;m \vdash_r^{\omega\cdot d,\alpha_0} \Gamma(n_0,\dots,n_l;m_0,\dots,m_k).$$

Proof.

We use induction over the height of derivation of Γ in $EA^1(I;O)$.

1. (C-Ax). Assume the derivation is an instance of the Σ_1-closure axiom where

$$\Gamma(\vec{x};\vec{a}) :\equiv \Gamma'(\vec{x};\vec{c}), A(\vec{d},\vec{b})$$

for $\vec{a}=\vec{c},\vec{d},\vec{b}$ and Σ_1-formula A. Thus in $EA(I;O)$ we have a proof of $A(\vec{y};\vec{b})$ and hence by the embedding theorem on page 61 we deduce in $EA_\infty(I;O)$ that

$$n',m' \vdash_r^{\omega\cdot h} A(\vec{n'};\vec{m'})$$

for some fixed $h,r \in \mathbf{N}$ where $n' \geq \max(\vec{n'})$ and $m' \geq \max(\vec{m'})$. Then applying cut-elimination followed by the (C-Ax) rule in $EA^1_\infty(I;O)$ we find

$$0;\max(n',m') \vdash_0^{0,\alpha_0} A(\vec{n'},\vec{m'})$$

for some $\alpha_0 \in E(\omega)$. Applying weakening we find

$$n;m \vdash_0^{0,\alpha_0} \Gamma'(\vec{n};\vec{m''}), A(\vec{n'},\vec{m'})$$

where $n \geq \vec{n}$ and $m \geq \max(\vec{m''},\vec{n'},\vec{m'})$, which is what we required.

The cases for a logical axiom, disjunction, conjunction and universal quantification are all straightforward following the corresponding cases in the proof of 2.27 in Chapter 2. We illustrate the two remaining cases.

2. ($\exists$). Assume the deduction in $EA^1(I;O)$ results from applying the existential rule to $\Gamma', A(t(\vec{y};\vec{b}))$ where $\vec{y};\vec{b}$ may contain other variables in addition to any of those in $\vec{x};\vec{a}$. By the induction hypothesis, assigning 0 to any extraneous variables

$$n; m \vdash_{r'}^{\omega\cdot d_0,\alpha_0} \Gamma', A(t(\vec{n};\vec{m})). \tag{4.9}$$

where $n \geq \vec{n}$ and $m \geq \vec{m}$ and $\alpha_0 \in E(\omega)$. Letting k be the value of $t(\vec{n};\vec{m})$ we deduce from logic axioms and cut, for some $d_1 \in \mathbf{N}$,

$$n; m \vdash_{1}^{\omega\cdot d_1,\alpha_0} \neg A(t(\vec{n};\vec{m})), A(k).$$

Letting $r := \max(r', |A|, 1)$ and $d' := \max(d_0, d_1) + 1$, a cut with (4.9) gives

$$n; m \vdash_{r}^{\omega\cdot d',\alpha_0} \Gamma', A(k).$$

Lemma 4.16 and Lemma 4.17 yield the computation $n; m \Vdash^{\omega\cdot(|t|+1),\alpha_0} k$. Hence, putting $d'' := \max(d', |t| + 1) + 1$ and $d := d'' + 1$, we have $\omega \cdot (|t| + 1) \in \omega \cdot d''[n]$ and $\omega \cdot d'' \in \omega \cdot d[n]$. We apply the existential rule and find

$$n; m \vdash_{r}^{\omega\cdot d,\alpha_0} \Gamma', \exists a A(a).$$

3. (P-Ind.). Assume $\Gamma :\equiv \Gamma', A(x)$ where the final rule of inference applied in $EA^1(I;O)$ is predicative induction. Applying the induction hypothesis to the premises gives

$$n; m \vdash_{r_0}^{\omega\cdot d_0,\beta_0} \Gamma', A(0)$$

and for every $m' \in \mathbf{N}$

$$n; \max(m, m') \vdash_{r_1}^{\omega\cdot d_1,\beta_0'} \Gamma', \neg A(m'), A(m' + 1).$$

Firstly we apply weakening (4.18) to replace β_0 and β_0' by $\alpha_0 := \beta_0 + \beta_0'$. Then by k-many applications of (L-Cut) on the formulae $A(0), A(1), \ldots, A(k - 1)$ we find for any k

$$n; \max(m, k) \vdash_{r}^{\omega\cdot d'+k,\alpha_0} \Gamma, A(k). \tag{4.10}$$

where $d' := \max(d_0, d_1)$ and $r := \max(r_0, r_1, |A|)$.

Using Lemma 4.17 and Lemma 4.16 we have $n; m \Vdash^{\omega,\alpha_0} k$. Thus by the (C-Cut) rule, with $d := d' + 1$

$$n; m \vdash_r^{\omega\cdot d,\alpha_0} \Gamma, A(k)$$

provided $k \leq n$ which is what we required.

□

Lemma 4.22. *Assume that in* $EA_\infty(I;O)$*, for a* Σ_1*-set* Δ*, we have a derivation*

$$n; m \vdash_0^{\alpha_1,\alpha_0} \Delta.$$

Further assume that $|t|$ *is the maximum of the heights of any term* t *in* Δ *and let* $\alpha'_i = \omega \cdot (|t| + 1) + \alpha_i$ *for* $i = 0, 1$*. Then* Δ *is true at* $B_{\alpha'_1,\alpha'_0}(n; m)$*.*

Proof.

Using induction on the height of the derivation, the (C-Ax) case follows the reasoning given for the (C-Ax) in Lemma 3.20. Then each of the other cases are given by replicating the proof of the corresponding result for $EA_\infty(I;O)$ with α_1 in place of α and α_0 suppressed.

□

Theorem 4.23.

$$ProvRec(EA^1(I;O)) = \mathcal{E}_1^4$$

Proof.

For the left to right inclusion we follow the method in the corresponding results 2.30 and 3.21 for $EA(I;O)$ and EA^1 respectively. Given a provably recursive function we take the Σ_1 existence proof and embed it into $EA^1_\infty(I;O)$. Then by the results above, for some $\alpha_1, \alpha_0 \in E(\omega)$, we find the function $B_{\alpha_1,\alpha_0}(n;m)$ witnesses the existential quantifiers.

Thus f is elementarily definable from $B_{\alpha_1,\alpha_0}(n;m)$ which by Lemma 4.13 is a function in $\mathcal{E}_1^4$. Therefore $f \in \mathcal{E}_1^4$. The right to left inclusion is Theorem 4.6.

$\square$

4.3.2 An Infinitary Theory for EA^2

We shall conclude this chapter by using another infinitary system in order to prove $ProvRec(EA^2) \subseteq \mathcal{E}^4$.

Definition 4.24. ***The infinitary theory*** EA^1_∞.

EA^2_∞ is defined analogously to EA^1_∞ from Definition 3.7 on page 79. EA^2_∞ will now employ a finite parameter d and two tree-ordinals, α_1 and α_0, to measure height. The height is again considered as a 'composition' $(d \circ (\alpha_1 \circ \alpha_0))$ or 'α_0 then α_1 then d'. We shall write $\vec{\alpha}$ for α_1, α_0. Sequents in EA^1_∞ take the form

$$m \vdash_r^{d,\vec{\alpha}} \Gamma$$

where m is a natural number declaration from the output domain. These sequents are inductively defined by the rules (C-Ax), (L-Ax), $\vee$, $\wedge$, $\forall$, $\exists$, (L-Cut) and (C-Cut) with computations rules again called ($\mathbb{O}$-Ax) and ($\mathbb{O}$-Cut). We shall only detail the definitions for the rules (C-Ax) and ($\mathbb{O}$-Ax). In each of the other rules the parameters $\vec{\alpha}$ act passively with the proof height determined by d similarly to Definition 3.7. The (C-Ax) rule now brings in derivations from $EA^1_\infty(I;O)$ whilst ($\mathbb{O}$-Ax) asserts a computation of a numeral bounded by the functions B_{α_1,α_0} used in $EA_\infty(I;O)$.

(C-Ax) $\max(n_0, m_0) \vdash_0^{d,\vec{\alpha}} \Gamma$ for any d, if in $EA^1_\infty(I;O)$ we already have a derivation of $n_0; m_0 \vdash_0^{\vec{\alpha}} \Gamma$.

($\mathbb{O}$-Ax) $\max(n_0, m_0) \Vdash^{d,\vec{\alpha}} k$ for any d, if $k \leq B_{\alpha_1,\alpha_0}(n_0; m_0)$.

Definition 4.25. *For* $\vec{\alpha} \in \Omega^S$ *and* $d \in \mathbf{N}$ *the functions* $B_{d,\vec{\alpha}} : \mathbf{N} \to \mathbf{N}$ *are defined by recursion over* d*:*

$$\begin{aligned} B_{0,\vec{\alpha}}(m) &:= B_{\vec{\alpha}}(m; m), \\ B_{d+1,\vec{\alpha}}(m) &:= B_{d,\vec{\alpha}}(B_{d,\vec{\alpha}}(m)). \end{aligned}$$

We immediately deduce from Lemma 4.12 that for a fixed $d \in \mathbf{N}$ and $\vec{\alpha} \in E(\omega)$, the function $B_{d,\vec{\alpha}}(m)$ is in $\mathcal{E}^4$.

Lemma 4.26. *For any* $\vec{\alpha} \in \Omega^S$ *and* $d, m \in \mathbf{N}$*,* $B_{d,\vec{\alpha}}(m)$ *is strictly increasing in* m *and* d*.*

Proof.

These majorization properties are provided by simple inductions over d using Lemma 4.14.

□

Lemma 4.27. ***Bounding for*** EA^2_∞***.*** *For any* $\vec{\alpha} \in \Omega^S$

$$m \Vdash^{d,\vec{\alpha}} k \qquad \Leftrightarrow \qquad k \leq B_{d,\vec{\alpha}}(m).$$

Proof.

Then the bounding result follows using the same argument as Lemma 3.11 from Chapter 3 with $\vec{\alpha}$ in place of α.

□

Lemma 4.28. ***Weakening for Computations.*** *If we have*

$$m \Vdash^{d,\vec{\alpha}} k$$

and if $m \leq m', k \geq k'$ *and* $d \leq d', \alpha_1[m] \subseteq \alpha_1'[m], \alpha_0[m] \subseteq \alpha_0'[m]$ *then for any* $\gamma, \delta \in \Omega^S$ *then we also have the computation*

$$m' \Vdash^{d',\vec{\alpha'}} k'$$

where $\vec{\alpha'} = \gamma + \alpha_1', \delta + \alpha_0'$.

Proof.

Using induction, if the computation is an ($\mathbb{O}$-Ax) the result follows using Lemma 4.16. Otherwise the computation results from an application of ($\mathbb{O}$-Cut). Then we simply apply the induction hypothesis and re-apply the rule.

□

Similarly using Lemma 4.17 we may deduce a bounding result for the values of terms. If the value of t when applied to $\vec{m} \leq m$ is k then $m \Vdash^{0,\vec{\alpha}} k$ where $\vec{\alpha} := \omega \cdot (|t| + 1), 0$.

It is now a matter of routine to show EA^2_∞ possesses cut-elimination by giving proofs of weakening, inversions and cut-rank reduction. All of these results are verified analogously to the proofs in Chapter 3 with $\vec{\alpha}$ in place of α. The base cases involving (C-Ax) now appeal to the corresponding results for $EA^1_\infty(I;O)$ given earlier in this chapter.

Theorem 4.29. ***Embedding of*** EA^2. *Assume*

$$EA^2 \vdash \Gamma(a_0, \ldots, a_k)$$

where all the free variables are displayed. Then this derivation determines some $\alpha_1, \alpha_0 \in E(\omega)$ *and some* $d, r \in \mathbf{N}$ *such that, for all* $m_0, \ldots, m_k$*, if* $m \geq \max(m_0, \ldots, m_k)$ *then*

$$m \vdash^{d,\vec{\alpha}}_r \Gamma(m_0, \ldots, m_k).$$

Proof.

We use induction over the proof of Γ in EA^2. The cases for logical axioms, disjunction, conjunction, universal quantification and cut are all straightforward following the method in 3.19. We shall provide more detail on the remaining two cases.

1. (C-Ax). Where $\Gamma(\vec{a}) :\equiv \Gamma'(\vec{c}), A(\vec{d}, \vec{b})$ for a Σ_1-formula A and the last rule is the closure axiom we have $EA^1(I;O) \vdash A(\vec{y}; \vec{b})$. Then, by the embedding theorem (4.21) and cut-elimination (4.20) for $EA^1_\infty(I;O)$, we obtain for $\alpha_1, \alpha_0 \in E(\omega)$

$$n; m \vdash_0^{\vec{\alpha}} A(n_0, \ldots, n_l, m_0, \ldots, m_j).$$

The result follows by applying (C-Ax) in EA^2_∞ and then weakening.

2. ($\exists$). Assume $\Gamma :\equiv \Gamma', A(t(\vec{b}))$ and the final rule of inference is existential quantification with witness t. Inductively we obtain

$$m \vdash_{r'}^{d_0, \vec{\alpha'}} \Gamma', A(t(\vec{m}))$$

for $\vec{\alpha'} := \alpha'_1, \alpha'_0 \in E(\omega)$. Assuming k is the value of $t(\vec{m})$ and putting $r := \max(r', |A|)$, we have by logic axioms and cuts

$$m \vdash_r^{d_1, \vec{\alpha'}} \Gamma', A(k) \tag{4.11}$$

for some $d_1 \in \mathbf{N}$. Letting $|t|$ be the height of the term t, we have a computation of k given by

$$m \Vdash^{0, \omega\cdot(|t|+1), 0} k.$$

Hence by computational weakening

$$m \Vdash^{0, \vec{\alpha}} k \tag{4.12}$$

where $\vec{\alpha} := \omega \cdot (|t| + 1) + \alpha'_1, \alpha'_0$. We also use weakening to change α'_1, α'_0 to $\vec{\alpha}$ in (4.11). Finally, with $d := d_1 + 1$ we may apply existential quantification using (4.12) to leave

$$m \vdash_r^{d, \vec{\alpha}} \Gamma.$$

□

Lemma 4.30. *Let Δ be a set of Σ_1-formulae. Assume that in EA^2_∞*

$$m \vdash_0^{d,\vec{\alpha}} \Delta.$$

Then letting $|t|$ be the maximum of the heights of any terms t in Δ and $\alpha'_i := \omega\cdot(|t|+1)+\alpha_i$ for $i = 0, 1$ we find Δ is true at $B_{d,\vec{\alpha}'}(m)$.

Proof.

We follow the proof of the analogous result for EA^1_∞ given by Lemma 3.20.

□

Theorem 4.31.

$$ProvRec(EA^2) = \mathcal{E}^4$$

Proof.

The right to left inclusion is already proved by Theorem 4.8. The left to right inclusion uses the same argument as before except now the witnesses for the existential formula are given by $B_{d,\vec{\alpha}}(m)$ which, for $\vec{\alpha} \in E(\omega)$ and fixed $d \in \mathbf{N}$, is a function in $\mathcal{E}^4$.

□

Chapter 5

The Hierarchy EA^k for $k \in \mathrm{N}$

5.1 Basic Definitions

We have two principles by which a given weak arithmetic base theory T with one type of variable may be extended. Firstly adding input constants to form a two-sorted theory $T(I;O)$ incorporating a predicative induction rule. Secondly defining a new one sorted theory T' by interpreting the input constants in $T(I;O)$ as proper variables when they occur in a Σ_1-formula via the (C-Ax) rule. This chapter presents a hierarchy of theories built up from alternating these two principles. The overall closure under finite applications of such principles gives a characterization of primitive recursive arithmetic.

Definition 5.1.

Let $EA(I;O)$ be denoted $EA^0(I;O)$. Then for any natural number $k > 0$ the theories EA^k and $EA^k(I;O)$ are generated inductively.

EA^k is a one-sorted theory with the usual rules of inference for first order logic with equality and the arithmetic axioms laid out on page 13. EA^k has no induction rule. We add one non-logical axiom schema:

The Σ_1**-closure axiom** of EA^k is

$$\text{(C-Ax)} \qquad EA^k \vdash \Gamma(\vec{c}), A(\vec{a}, \vec{b}) \qquad \text{if} \quad EA^{k-1}(I;O) \vdash A(\vec{x}; \vec{b}).$$

where A is a Σ_1-formula, Γ is an arbitrary set of formulae, and where all the free variables of Γ, A are indicated.

$EA^k(I;O)$ is then defined as the two sorted extension of EA^k. That is, we add an infinite supply of input constants $x, y, z, x_0, x_1, \ldots$ as symbols of the language along with the predicative induction rule:

The **predicative induction** rule is, for an arbitrary set of formulae Γ

$$\text{(P.Ind.)} \qquad \frac{\Gamma, A(0) \qquad \Gamma, \neg A(a), A(a+1)}{\Gamma, A(x)} \qquad \text{where } a \text{ is not free in } \Gamma.$$

5.2 Lower Bounds for Provably Recursive Functions

Theorem 5.2. *For each natural number $k > 0$*

$$ProvRec(EA^k) \supseteq \mathcal{E}^{k+2}.$$

Proof.

We use induction over k. If $k := 1$ then we use Proposition 3.3 in Chapter 3. Now assume the result holds for k. Then using Fact 4.7, for any function $f \in \mathcal{E}^{k+3}$ there are three possibilities for its definition. We use a sub-induction over the three cases.

1. $f \in \mathcal{E}^{k+2}$. Then f is provably recursive in EA^k by the induction hypothesis for k. As $ProvRec(EA^k) \subseteq ProvRec(EA^{k+1})$ we see f is provably recursive in EA^{k+1}.

2. Now assume f is defined by composition where the auxiliary functions g_i are in $\mathcal{E}^{k+3}$. By the sub-induction hypothesis we have $g_i \in ProvRec(EA^{k+1})$. Then we may use Theorem 3.4 to show $ProvRec(EA^{k+1})$ is closed under composition and hence, $f \in ProvRec(EA^{k+1})$.

3. Finally it may be the case that $f \in \mathcal{E}^{k+3}$ is defined by a primitive recursion using auxiliary functions $g_i \in \mathcal{E}^{k+2}$. Using the main induction hypothesis $g_i \in ProvRec(EA^k)$. Then using Theorem 4.3 we may show $f \in ProvRec(EA^k(I; O))$ and thus we obtain $f \in ProvRec(EA^{k+1})$.

$\square$

Remarks

We note that the previous theorem may also be proved using the relationship between bounded arithmetic and EA^1 illustrated by Proposition 3.6. Let f_k for $k \in \mathbf{N}$ be some suitable diagonal function for the Grzegorczyk Hierarchy where f_2 is elementary. For example one may take the finite levels of the Fast-Growing Hierarchy defined in Appendix B. Then the provably recursive functions of $I\Delta_0$ plus the axiom "f_{k+2} is total" are exactly $\mathcal{E}^{k+3}$. This result is given in [12] where it is attributed to A. Wilkie. The function f_2 is provably recursive in EA^1 and, inductively, Theorem 4.3 yields that f_{k+2} is provably recursive in EA^{k+1}. Following the approach in the proof of 3.6 we may embed $I\Delta_0+$ "f_{k+2} is total" into EA^{k+1}. Hence $ProvRec(EA^k) \supseteq \mathcal{E}^{k+2}$.

5.3 Infinitary Theories for EA^k

To provide matching upper bounds on the provably recursive functions of EA^k we must define suitable infinitary theories. We first give a definition of the intended bounding functions these infinitary theories will use. In what follows let $\vec{\alpha} = \alpha_{k-1}, \ldots, \alpha_0$ for any natural number $k > 0$.

Definition 5.3. *Recall the definition of* $B_{\alpha_0}(n;m)$ *given in 2.11. For* $\vec{\alpha} \in \Omega^S$ *and* $k > 0$*, the functions* $B_{\alpha_k,\vec{\alpha}} : \mathbf{N}^2 \to \mathbf{N}$ *are defined by recursion on* α_k*:*

$$\begin{aligned} B_{0,\vec{\alpha}}(n;m) &:= B_{\vec{\alpha}}(m;m), \\ B_{\alpha_k+1,\vec{\alpha}}(n;m) &:= B_{\alpha_k,\vec{\alpha}}(n;B_{\alpha_k,\vec{\alpha}}(n;m)), \\ B_{\lambda,\vec{\alpha}}(n;m) &:= B_{\lambda_n,\vec{\alpha}}(n;m). \end{aligned}$$

For $d \in \mathbf{N}$ *we also define the functions* $B_{d,\vec{\alpha}} : \mathbf{N} \to \mathbf{N}$ *by recursion on* d*:*

$$\begin{aligned} B_{0,\vec{\alpha}}(m) &:= B_{\vec{\alpha}}(m;m), \\ B_{d+1,\vec{\alpha}}(m) &:= B_{d,\vec{\alpha}}(B_{d,\vec{\alpha}}(m)). \end{aligned}$$

We may easily verify, by an induction on α_k, that $B_{\alpha_k,\vec{\alpha}}(n;m) = B_{d,\vec{\alpha}}(m)$ where $d := G_{\alpha_k}(n)$.

Lemma 5.4. *If* $\vec{\alpha} \in E(\omega)$ *and* $k > 0$ *then*

1. For a fixed $d \in \mathbf{N}$ *we have* $B_{d,\vec{\alpha}} \in \mathcal{E}^{k+2}$*,*

2. For $\alpha_k \in E(\omega)$ *we have* $B_{\alpha_k,\vec{\alpha}} \in \mathcal{E}^{k+3}$*.*

Proof.

We use induction over k. When $k := 1$ we know $B_{d,\vec{\alpha}} \in \mathcal{E}^3$ by 3.9 and $B_{\alpha_1,\vec{\alpha}} \in \mathcal{E}^4$ by 4.12.

Now, assuming the result for k, it is easy to see that $B_{d,\alpha_k,\vec{\alpha}}$ is defined by composition using $B_{\alpha_k,\vec{\alpha}} \in \mathcal{E}^{k+3}$. Hence it is also a function in $\mathcal{E}^{k+3}$.

From above, $B_{\alpha_{k+1},\alpha_k,\vec{\alpha}}(n;m)$ is equal to $B_{d,\alpha_k,\vec{\alpha}}(m)$ where $d := G_{\alpha_{k+1}}(n)$. Since $G_{\alpha_{k+1}}(n)$ is always elementary for $\alpha_{k+1} \in E(\omega)$ we may define $B_{\alpha_{k+1},\alpha_k,\vec{\alpha}}$ by a primitive recursion whose auxiliary functions lie in $\mathcal{E}^{k+3}$. We immediately conclude, as required, $B_{\alpha_{k+1},\alpha_k,\vec{\alpha}} \in \mathcal{E}^{k+4}$.

□

Definition 5.5.

The theories EA^k_∞ and $EA^k_\infty(I;O)$ for any natural number $k > 0$ are generated inductively by generalizing the definitions given in Chapter 3 and Chapter 4.

Let $EA_\infty(I;O)$ be denoted $EA^0_\infty(I;O)$. Sequents in $EA^k_\infty(I;O)$ take the form

$$n; m \vdash^{\alpha_k, \vec{\alpha}}_r \Gamma.$$

The rules of inference for (C-Ax) and ($\mathbb{O}$-Ax) are given below. The other rules are standard with the conclusions being assigned a height $\alpha_k, \vec{\alpha}$ provided β belongs to $\alpha_k[n]$ whilst $\vec{\alpha}$ remains fixed.

(C-Ax) $\quad n; \max(n_0, m_0) \vdash^{\alpha_k, \vec{\alpha}}_0 \Gamma \quad$ for any α_k and n, if in $EA^{k-1}_\infty(I;O)$ we already have $n_0; m_0 \vdash^{\vec{\alpha}}_0 \Gamma$.

($\mathbb{O}$-Ax) $\quad n; \max(n_0, m_0) \Vdash^{\alpha_k, \vec{\alpha}} m' \quad$ for any α_k and n, if $m' \leq B_{\vec{\alpha}}(n_0; m_0)$.

The theory EA^1_∞ is defined in Chapter 3. The theories EA^{k+1}_∞ for $k > 0$ are defined similarly. That is, they are the same system as $EA^{k+1}_\infty(I;O)$ but without the input parameter n and using a finite measure d in place of the tree-ordinal α_{k+1}.

Theorem 5.6. *For each natural number $k > 0$*

$$ProvRec(EA^k) = \mathcal{E}^{k+2}.$$

Proof.

Using the infinitary theory EA^k_∞ we follow the methods laid out in Chapter 3, where $k := 1$, and Chapter 4, where $k := 2$. We may prove an embedding of EA^k into EA^k_∞ and find witnesses for the provable Σ_1-formulae of EA^k now given by the functions $B_{d,\vec{\alpha}}(m)$. Since these functions lie in $\mathcal{E}^{k+2}$ by Lemma 5.4, we find $ProvRec(EA^k) \subseteq \mathcal{E}^{k+2}$. The reverse inclusion is Theorem 5.2.

□

5.4 The Theory $EA^{\prec\omega}$

Definition 5.7. *$EA^{\prec\omega}$ is defined as the two-sorted theory which extends $EA(I;O)$ with the following Σ_1-closure axiom schema:*

(C-Ax) $$EA^{\prec\omega} \vdash \Gamma(\vec{c}), A(\vec{a}, \vec{b}) \qquad \textit{if} \quad EA^{\prec\omega} \vdash A(\vec{x}; \vec{b}).$$

where A is a Σ_1-formula, Γ is an arbitrary set of formulae, and where all the free variables of Γ, A are indicated.

Thus $EA^{\prec\omega}$ is closed under the two principals of predicative induction and Σ_1-closure.

It is easily seen that $EA^{\prec\omega}$ is closed under Σ_1-induction. For any Σ_1-formula $A(a)$ which is progressive in a, predicative induction gives $A(x)$. Then Σ_1-closure and quantification yield $\forall a A(a)$.

Theorem 5.8.

$$ProvRec(EA^{\prec\omega}) = PRIM.$$

Proof.

Working from left to right if f is provably recursive in $EA^{\prec\omega}$ then by parsing the finite proof of the existence condition we will observe a finite number of appeals to (C-Ax) and (P.Ind.). Thus the proof may be replicated in EA^k for some $k \in \mathbf{N}$ and by 5.6 we deduce $f \in \mathcal{E}^{k+2} \subseteq PRIM$.

For the right to left inclusion note that if $f \in PRIM$ then for some $k \in \mathbf{N}$ we have $f \in \mathcal{E}^{k+2}$. Then by 5.6 f is provably recursive in EA^k and hence provably recursive in $EA^{\prec\omega}$.

□

Appendix

A. Derivations of Basic Results

In the following lemmas Γ is an arbitrary set of formulae.

Lemma A.1. ***Generalized law of excluded middle.*** *For any formula A*

$$EA(I;O) \vdash \Gamma, \neg A, A.$$

Proof.

We use induction over the build up of the formula A. We only need consider the atomic case $s = t$ and the inductive steps $(\vee)$ and $(\exists)$ due to the symmetry of the calculus and free use of exchange. If $A :\equiv s = t$ then the result is an instance of excluded middle. For the inductive step $A :\equiv A_0 \vee A_1$ we have

$$\begin{array}{c} \dfrac{\dfrac{\overset{\text{[IH]}}{\Gamma, \neg A_0, A_0} \qquad \overset{\text{[IH]}}{\Gamma, \neg A_1, A_1}}{\Gamma, \neg A_0 \wedge \neg A_1, A_0, A_1}\,(\wedge)}{\Gamma, \neg A_0 \wedge \neg A_1, A_0 \vee A_1}\,(\vee) \\ \hdashline \Gamma, \neg A, A. \end{array}$$

Finally when $A :\equiv \exists a(A_0(a))$ we have

$$\begin{array}{c} \dfrac{\dfrac{\overset{\text{[IH]}}{\Gamma, \neg A_0(a), A_0(a)}}{\Gamma, \neg A_0(a), \exists a(A_0(a))}\,(\exists)}{\Gamma, \forall a(\neg A_0(a)), \exists a(A_0(a))}\,(\forall) \\ \hdashline \Gamma, \neg A, A. \end{array}$$

□

Henceforth we use the lemma above as an axiom denoted by (L-Ax).

Lemma A.2. ***Conjunction inversion.*** *For any formulae A_0 and A_1,*

$$EA(I;O) \vdash \Gamma, A_0 \wedge A_1 \qquad \Rightarrow \qquad EA(I;O) \vdash \Gamma, A_i$$

where $i = 0$ or 1.

Proof.

Without loss of generality let $i = 0$. Assume that we have a derivation of $\Gamma, A_0 \wedge A_1$, then

$$\cfrac{\begin{array}{cl}[\text{L-Ax}] & \\ \Gamma, \neg A_1, \neg A_0, A_0 & \\ \hline \Gamma, A_0, \neg A_0 \vee \neg A_1 & (\vee) \\ \hdashline \Gamma, A_0, \neg(A_0 \wedge A_1) & \end{array} \qquad \begin{array}{c}[\text{Assumption}] \\ \Gamma, A_0, A_0, \wedge A_1\end{array}}{\Gamma, A_0.}\ (\text{Cut})$$

□

Lemma A.3. ***Disjunction inversion.*** *For any formulae A_0 and A_1,*

$$EA(I;O) \vdash \Gamma, A_0 \vee A_1 \qquad \Rightarrow \qquad EA(I;O) \vdash \Gamma, A_0, A_1.$$

Proof.

Assume we have a derivation of $\Gamma, A_0 \vee A_1$, then

$$\cfrac{\begin{array}{cl}[\text{L-Ax}] \qquad\qquad [\text{L-Ax}] & \\ \Gamma, A_1, \neg A_0, A_0 \qquad \Gamma, A_0, \neg A_1, A_1 & \\ \hline \Gamma, A_0, A_1, \neg A_0 \wedge \neg A_1 & (\wedge) \\ \hdashline \Gamma, A_0, A_1, \neg(A_0 \vee A_1) & \end{array} \qquad \begin{array}{c}[\text{Assumption}] \\ \Gamma, A_0, A_1, A_0 \vee A_1\end{array}}{\Gamma, A_0, A_1.}\ (\text{Cut})$$

□

Lemma A.4. ***Universal quantifier inversion.*** *For any formulae A,*

$$EA(I;O) \vdash \Gamma, \forall a A(a) \qquad \Rightarrow \qquad EA(I;O) \vdash \Gamma, A(t)$$

where t is any term.

Proof.

Assuming a derivation of $\Gamma, \forall a A(a)$ we deduce

$$\dfrac{\begin{array}{c}[\text{L-Ax}]\\ \dfrac{\Gamma, \neg A(t), A(t)}{\Gamma, \exists a(\neg A(a)), A(t)}\,(\exists)\\ \cdots\cdots\cdots\cdots\cdots\cdots\\ \Gamma, \neg\forall a(A(a)), A(t)\end{array} \qquad \begin{array}{c}[\text{Assumption}]\\ \Gamma, \forall a A(a)\end{array}}{\Gamma, A(t).}\,(\text{Cut})$$

□

We use inversions in proof-trees as derived rules in their own right with the notations ($\wedge$-inv), ($\vee$-inv) and ($\forall$-inv) respectively. They may be used to give more liberal equality and arithmetic axioms such as $\Gamma, s \neq t, t \neq w, w = s$. We still denote such sequents as [E-Ax] or [Ax].

Lemma A.5. ***Symmetry of equality.*** *For any terms s and t,*

$$EA(I;O) \vdash \Gamma, s \neq t, t = s.$$

Proof.

$$\dfrac{\begin{array}{c}[\text{E-Ax}]\\ \Gamma, s \neq t, t \neq t, t = s\end{array} \qquad \begin{array}{c}[\text{E-Ax}]\\ \Gamma, t = t\end{array}}{\Gamma, s \neq t, t = s.}\,(\text{Cut})$$

□

Lemma A.6. ***Transitivity of equality.*** *For any terms s, t and w,*

$$EA(I;O) \vdash \Gamma, s \neq t, t \neq w, s = w.$$

Proof.

$$\dfrac{\begin{array}{c}[\text{E-Ax}]\\ \Gamma, s \neq t, t \neq w, w = s\end{array} \qquad \begin{array}{c}[\text{Lemma A.5}]\\ \Gamma, w \neq s, s = w\end{array}}{\Gamma, s \neq t, t \neq w, s = w.}\,(\text{Cut})$$

□

Lemma A.7. ***Generalized law of equality.*** *For any terms s, t and formula A,*

$$EA(I;O) \vdash \Gamma, s \neq t, \neg A(s), A(t).$$

Proof.

We use induction over the build up of the formula $A(t)$. By symmetry we only need consider the atomic cases $w = t, t = w$ and the inductive steps $(\vee)$ and $(\exists)$. If $A :\equiv w = t$ then the result is an instance of transitivity of equality given above. Now let $A :\equiv t = w$. We have

$$\dfrac{\overset{\text{[E-Ax]}}{\Gamma, w \neq s, s \neq t, t = w} \qquad \overset{\text{[Lemma A.5]}}{\Gamma, s \neq w, w = s}}{\Gamma, s \neq t, s \neq w, t = w.}\ \text{(Cut)}$$

If $A(t) :\equiv A_0(t) \vee A_1(t)$ we may deduce

$$\begin{array}{c} \overset{\text{[IH]}}{\Gamma, s \neq t, \neg A_0(s), A_0(t)} \qquad \overset{\text{[IH]}}{\Gamma, s \neq t, \neg A_1(s), A_1(t)} \\ \hline \Gamma, s \neq t, \neg A_0(s) \wedge \neg A_1(s), A_0(t), A_1(t) \\ \hline \Gamma, s \neq t, \neg A_0(s) \wedge \neg A_1(s), A_0(t) \vee A_1(t) \\ \hdashline \Gamma, s \neq t, \neg A(s), A(t). \end{array} \begin{array}{l} (\wedge) \\ (\vee) \end{array}$$

Lastly if $A(t) :\equiv \exists a(A_0(a,t))$ we have

$$\begin{array}{c} \overset{\text{[IH]}}{\Gamma, s \neq t, \neg A_0(a,s), A_0(a,t)} \\ \hline \Gamma, s \neq t, \neg A_0(a,s), \exists a(A_0(a,t)) \\ \hline \Gamma, s \neq t, \forall a(\neg A_0(a,s)), \exists a(A_0(a,t)) \\ \hdashline \Gamma, s \neq t, \neg A(s), A(t). \end{array} \begin{array}{l} (\exists) \\ (\forall) \end{array}$$

□

Lemma A.8. ***Substitution.*** *For any terms s and t,*

$$EA(I;O) \vdash \Gamma, s = t \quad \textit{and} \quad EA(I;O) \vdash \Gamma, A(s) \qquad \Rightarrow \qquad EA(I;O) \vdash \Gamma, A(t).$$

Proof.

$$\dfrac{\dfrac{\overset{\text{[Assumption]}}{\Gamma, s = t} \qquad \overset{\text{[Lemma A.7]}}{\Gamma, s \neq t, \neg A(s), A(t)}}{\Gamma, \neg A(s), A(t)}\ \text{(Cut)} \qquad \overset{\text{[Assumption]}}{\Gamma, A(s)}}{\Gamma, A(t)}\ \text{(Cut)}$$

□

Lemma A.9. ***Cases.*** *For any term t, if a is not free in Γ,*

$$EA(I;O) \vdash \Gamma, A(0) \quad \textit{and} \quad EA(I;O) \vdash \Gamma, A(a+1) \quad \Rightarrow \quad EA(I;O) \vdash \Gamma, A(t).$$

Proof.

Firstly we have

$$\dfrac{\overset{\text{[Assumption]}}{\Gamma, A(0)} \qquad \overset{\text{[Lemma A.7]}}{\Gamma, t \neq 0, \neg A(0), A(t)}}{\Gamma, t \neq 0, A(t).}\ \text{(Cut)}$$

Secondly

$$\dfrac{\dfrac{\overset{\text{[Assumption]}}{\Gamma, A(a+1)} \qquad \overset{\text{[Lemma A.7]}}{\Gamma, a+1 \neq t, \neg A(a+1), A(t)}}{\Gamma, a+1 \neq t, A(t)}\ \text{(Cut)}}{\Gamma, \forall a(a+1 \neq t), A(t).}\ (\forall)$$

Forming the conjunction of the two derivations above we obtain

$$\dfrac{\Gamma, t \neq 0 \wedge \forall a(a+1 \neq t), A(t) \qquad \dfrac{\dfrac{\dfrac{\overset{\text{[Ax (1.3)]}}{\Gamma, t \neq 0 \vee (t \dot{-} 1) + 1 = t}}{\Gamma, t \neq 0, (t \dot{-} 1) + 1 = t}\ (\vee\text{-inv})}{\Gamma, t \neq 0, \exists a(a+1 = t)}\ (\exists)}{\Gamma, t \neq 0 \vee \exists a(a+1 = t).}\ (\vee)}{\Gamma, A(t).}\ \text{(Cut)}$$

□

We use the leaf abbreviation [E-Ax] to refer to instances of Lemma A.5, Lemma A.6 or Lemma A.7. We use Lemma A.8 and Lemma A.9 as derived rules within proof-trees adopting the notation (Sub.) or (Cases) respectively.

Lemma A.10. *In $EA(I;O)$ we may prove the universal closures of:*

1. $a \leq a$.

2. $a \leq 0 \to a = 0$.

3. $a \leq b + 1 \to a \leq b \lor a = b + 1$.

4. $a \leq b \to a \leq b + 1$.

5. $a + 1 \leq b \to a \leq b$.

6. $a \leq b \to a + 1 \leq b + 1$.

7. $a \leq b \land a' \leq b' \to a + a' \leq b + b'$.

8. $a \leq b \land a' \leq b' \to a \cdot a' \leq b \cdot b'$.

9. $\neg 0 < a$.

10. $a < a + 1$.

11. $a < b \to a + 1 \leq b$.

12. $a < b \to a + 1 < b + 1$.

13. $a < b \to a < b + 1$.

14. $a < b + 1 \to a < b \lor a = b$.

Proofs.

1.

$$\dfrac{\dfrac{\text{[Ax (1.6)]}}{a + 0 = a}}{\dfrac{\exists c(a + c = a)}{\cdots\cdots\cdots\cdots\cdots\cdots}\; a \leq a.}\;(\exists)$$

2. Firstly

$$\dfrac{\overset{[\text{L-Ax}]}{a \neq 0, a = 0} \qquad \overset{[\text{Ax (1.6)}]}{a + 0 = a}}{a + 0 \neq 0, a = 0.}\ (\text{Sub.})$$

Secondly we have

$$\dfrac{\overset{[\text{Ax (1.1)}]}{(a + c) + 1 \neq 0, a = 0} \qquad \overset{[\text{Ax (1.7)}]}{a + (c + 1) = (a + c) + 1}}{a + (c + 1) \neq 0, a = 0.}\ (\text{Sub.})$$

Applying the cases lemma (A.9) to these derivations gives

$$\begin{array}{cl} a + c \neq 0, a = 0 & \\ \hline \forall c(a + c \neq 0), a = 0 & (\forall) \\ \hline \forall c(a + c \neq 0) \vee a = 0 & (\vee) \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\ a \leq 0 \to a = 0. & \end{array}$$

3. Following the lines of the deduction above for part 2

$$\dfrac{\overset{[\text{L-Ax}]}{a \neq b + 1, a = b + 1} \qquad \overset{[\text{Ax (1.6)}]}{a + 0 = a}}{a + 0 \neq b + 1, a = b + 1.}\ (\text{Sub.})$$

The second derivation is

$$\dfrac{\dfrac{\overset{[\text{Ax (1.1)}]}{(a + c) + 1 \neq b + 1, a + c = b}}{(a + c) + 1 \neq b + 1, \exists c(a + c = b)}\ (\exists) \qquad \overset{[\text{Ax (1.7)}]}{a + (c + 1) = (a + c) + 1}}{a + (c + 1) \neq b + 1, \exists c(a + c = b).}\ (\text{Sub.})$$

Applying cases we obtain

$$\begin{array}{cl} a + c \neq b + 1, \exists c(a + c = b), a = b + 1 & \\ \hline \forall c(a + c \neq b + 1), \exists c(a + c = b), a = b + 1 & (\forall) \\ \hline \forall c(a + c \neq b + 1), \exists c(a + c = b) \vee a = b + 1 & (\vee) \\ \hline \forall c(a + c \neq b + 1) \vee (\exists c(a + c = b) \vee a = b + 1) & (\vee) \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\ a \leq b + 1 \to a \leq b \vee a = b + 1. & \end{array}$$

4.

$$\dfrac{\dfrac{\dfrac{\dfrac{\overset{[\text{E-Ax}]}{a+c\neq b,(a+c)+1=b+1} \qquad \overset{[\text{Ax (1.7)}]}{a+(c+1)=(a+c)+1}}{a+c\neq b, a+(c+1)=b+1}\ (\text{Sub.})}{a+c\neq b,\exists c(a+c=b+1)}\ (\exists)}{\forall c(a+c\neq b),\exists c(a+c=b+1)}\ (\forall)}{a\leq b\rightarrow a\leq b+1.}$$

5. From the equality axiom

$$(a+1)+c\neq b,(a+1)+c=b$$

we use associativity of addition (1.14), to deduce by substitution

$$(a+1)+c\neq b, a+(1+c)=b+1.$$

The result follows by existential quantification on the right with witness $1+c$ and then universal quantification over the remaining c.

6. Along similar lines to part 5 we use the equality axiom

$$a+c\neq b,(a+c)+1=b+1$$

and then both associativity (1.13) and commutativity (1.14) of addition to deduce by substitutions

$$a+c\neq b,(a+1)+c=b+1.$$

Then we apply existential quantification on the right with witness c and universal quantification over the remaining c.

7. This time we start with the equality

$$a+c\neq b, a'+c'\neq b',(a+c)+(a'+c')=b+b'.$$

The associativity and commutativity axioms will give

$$a+c\neq b, a'+c'\neq b',(a+a')+(c+c')=b+b'.$$

An existential quantification at $c + c'$ followed by two universal quantifications, over c and c', complete the derivation.

8. Like the last three parts but with the equality

$$a + c \neq b, a' + c' \neq b', (a \cdot c) + (a' \cdot c') = b \cdot b'.$$

This time we make repeated appeals to the distributivity of addition over multiplication (1.15) and (1.16) as well as associativity and commutativity of addition until we reach

$$a + c \neq b, a' + c' \neq b', (a \cdot a) + ((a \cdot c') + ((a' \cdot c) + (c \cdot c'))) = b \cdot b'.$$

The existential witness for the inequality will now be the term $(a \cdot c') + ((a' \cdot c) + (c \cdot c'))$.

9. We have $\neg a \leq 0, a = 0$ using part 2. Written another way this is $\neg(a \leq 0 \wedge a \neq 0)$ which by the definition of $<$ is just $\neg 0 < a$.

10. Firstly, parts 1 and 4 give $a \leq a + 1$. Secondly, using equality and substitutions on axiom (1.10) we may show $a \neq a + 1, 0 = 0 + 1$ which by a cut with (1.1) is $a \neq a + 1$. Hence by conjunction $a \leq a + 1 \wedge a \neq a + 1$, that is $a < a + 1$.

11. We easily deduce $a+0 \neq b, a = b$ from [L-Ax] and (1.6). Furthermore, we have by [L-Ax], associativity of addition (1.13) and commutativity of addition (1.14) that $a+(c+1) \neq b, (a + 1) + c = b$. Hence by (∃), $a + (c + 1) \neq b, a + 1 \leq b$. Using the cases rule on these two derivations leaves $a + c \neq b, a = b, a + 1 \leq b$ which is $a < b \rightarrow a + 1 \leq b$ by a universal quantification over c and logic.

12. From part 6, $\neg a \leq b, a + 1 \leq b + 1$ and from axiom (1.2), $a + 1 \neq b + 1, a = b$. Forming the conjunct $\neg a \leq b, a = b, a+1 \leq b+1 \wedge a+1 \neq b+1$ and using the definition of $<$ we have shown $a < b \rightarrow a + 1 < b + 1$.

13. Using the equality axiom $b + (c + 1) \neq b, (b + (c + 1)) \dot{-} b = b \dot{-} b$ with substitution instances of axiom (1.10) gives $b + (c + 1) \neq b, c + 1 = 0$ from which we cut with $c + 1 \neq 0$ from axiom (1.1). Further substitutions using associativity and commutativity

of addition yields $(b+1)+c \neq b$ which allows a cut with the equality axiom $a \neq b+1, a+c \neq b, (b+1)+c = b$. We now have $a \neq b+1, a+c \neq b$ so by universal quantification $\neg a \leq b, a \neq b+1$. Forming a conjunction with an instance of part 4, $\neg a \leq b, a \leq b+1$, will give $\neg a \leq b, a < b+1$. The result follows by weakening since $\neg a \leq b, a = b, a < b+1$ is just $\neg a < b, a < b+1$.

14. By part 3 we find $\neg a \leq b+1, a \leq b, a = b+1$. Forming a conjunct with the axiom $a \neq b, a = b$ gives $\neg a \leq b+1, a = b+1, a \leq b \wedge a \neq b, a = b$. This is exactly $a < b+1 \rightarrow a < b \vee a = b$.

$\square$

B. Sub-Recursive Hierarchies

The definitions and facts we give here are taken from [43] and [51] where further details and proofs may be found.

Definition B.11. *Let $\vec{n} := n_1, \ldots, n_k \in \mathbf{N}$. The* ***initial functions*** $z, s, u : \mathbf{N}^k \rightarrow \mathbf{N}$ *are defined as*

$$\begin{aligned} z(\vec{n}) &:= 0, \\ s(\vec{n}) &:= n_1 + 1, \\ u^i(\vec{n}) &:= n_i, \quad \textit{for each } i \leq k. \end{aligned}$$

Definition B.12. *Given the j-ary function h and the k-ary functions $g_1, \ldots, g_j$ we may define a new function $f : \mathbf{N}^k \rightarrow \mathbf{N}$ by the scheme of* ***composition*** *as*

$$f(\vec{n}) := h(g_1(\vec{n}), \ldots, g_j(\vec{n})).$$

Definition B.13. *Given the $k+2$-ary function h and the k-ary function g we may define a new function $f : \mathbf{N}^{k+1} \rightarrow \mathbf{N}$ by the scheme of* ***primitive recursion*** *as*

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

Furthermore, we say f is defined by ***bounded recursion*** *if for some previous defined $k+1$-ary function p*

$$\begin{cases} f(0,\vec{n}) & := & g(\vec{n}), \\ f(m+1,\vec{n}) & := & h(m,\vec{n},f(m,\vec{n}), \\ f(m,\vec{n}) & \leq & p(m,\vec{n}). \end{cases}$$

Definition B.14. *Given the $k+1$-ary function g, and the binary functions addition and multiplication, we may define a new function $f : \mathbf{N}^{k+1} \to \mathbf{N}$ by* ***bounded sum*** *as*

$$f(m,\vec{n}) \;:=\; \Sigma_{i\leq m} g(i,\vec{n}).$$

or by ***bounded product*** *as*

$$f(m,\vec{n}) \;:=\; \Pi_{i\leq m} g(i,\vec{n}).$$

Definition B.15. *The class of* ***primitive recursive functions****, $PRIM$, is defined as the smallest class of functions which contains the initial functions and is closed under composition and primitive recursion.*

Definition B.16. *(Csillag [17], Kalmar [32]) The class of* ***elementary functions****, $\mathcal{E}$, is defined as the smallest class of functions which contains the initial functions, the addition function, the recursive difference function and is closed under formation of bounded sums and products.*

Definition B.17. *For $i \in \mathbf{N}$ let E_i be a sequence of primitive recursive functions defined by:*

$$\begin{array}{lcl} E_0(n,m) & := & n+m, \\ E_1(n) & := & n^2+2, \\ E_{i+2}(0) & := & 2, \\ E_{i+2}(n+1) & := & E_{i+1}(E_{i+2}(n)). \end{array}$$

It can be seen easily that the functions E_i are monotonically increasing in their numerical input and also in their numerical index. Furthermore, if we denote by E_i^k the k-times iterate of E_i, we have that $E_i^k(n) \leq E_{i+1}(n+k)$ (cf. [51]).

Definition B.18. *(Grzegorczyk [29]) The function class $\mathcal{E}^0$ is defined as the smallest class of functions which contains the initial functions and is closed under composition and bounded recursion. The function classes $\mathcal{E}^{i+1}$ for any $i \geq 0$ are defined similarly except the function E_i is added to the list of initial functions.*

These classes form the ***Finite Grzegorczyk Hierarchy****, $\mathcal{E}^0, \mathcal{E}^1, \ldots, \mathcal{E}^i, \ldots$ where $i \in \mathbf{N}$.*

Facts B.19. *The proofs of the following results regarding the Grzegorczyk Hierarchy may be found in [51].*

1. $\bigcup_{i \in \mathbf{N}} \mathcal{E}^i = PRIM$.

2. $\mathcal{E}^3 = \mathcal{E}$, *the elementary functions.*

3. $\mathcal{E}^0 \subset \mathcal{E}^1 \subset \ldots \subset \mathcal{E}^i \subset \mathcal{E}^{i+1} \subset \ldots$.

4. $f(\vec{n}) \in \mathcal{E}^0 \Rightarrow f(\vec{n}) \leq n_j + c$ *for some* $c \in \mathbf{N}$ *and* $j \leq k$.

5. $f(\vec{n}) \in \mathcal{E}^1 \Rightarrow f(\vec{n}) \leq n_k \cdot c_k + \ldots n_0 \cdot c_0 + c$ *for some* $c, c_0, \ldots, c_k \in \mathbf{N}$.

6. $f(\vec{n}) \in \mathcal{E}^2 \Rightarrow f(\vec{n}) \leq p(\vec{n})$ *where* p *is some polynomial in* $\vec{n}$.

7. $f(\vec{n}) \in \mathcal{E}^3 \Rightarrow f(\vec{n}) \leq 2_m(\vec{n})$ *for some* $m \in \mathbf{N}$
where $2_0(\vec{n}) := max(\vec{n})$ *and* $2_{m+1}(\vec{n}) := 2^{2_m(\vec{n})}$.

8. $f(\vec{n}) \in \mathcal{E}^{i+1} \Rightarrow f(\vec{n}) \leq E_i^m(max(\vec{n}))$ *for some* $m \in \mathbf{N}$.

9. If $f(\vec{n}) \in \mathcal{E}^i$ *then fixed iterates of* f *belong to the same class by composition whereas the full iteration* $f'(m, \vec{n}) := f^m(\vec{n})$ *lies in the next class* $\mathcal{E}^{i+1}$.

10. If $f(\vec{n})$ *is defined by primitive recursion using functions* $g, h \in \mathcal{E}^i$ *then* $f \in \mathcal{E}^{i+1}$.

Definition B.20. *Let f be any k-ary computable function and let $\mathcal{C}$ be any recursively enumerable class of functions. We say f is* ***computable in time belonging to $\mathcal{C}$*** *if there* *is a k-ary function $s_f \in \mathcal{C}$ such that $s_f(\vec{n})$ gives the number of steps required by a deterministic Turing machine to compute f on inputs $\vec{n}$. Likewise we may define the notion f is computable in* ***space belonging to*** *$\mathcal{C}$.*

Fact B.21. *(Cobham [15], Meyer and McCreight [40], Meyer [41]) Let $i \geq 3$ and let f be any total function. Then the following are equivalent*

1. *f belongs to the Grzegorczyk class $\mathcal{E}^i$.*
2. *f is computable in time belonging to $\mathcal{E}^i$.*
3. *f is computable in space belonging to $\mathcal{E}^i$.*

Hence $\mathcal{E}^i$ forms a complexity class with respect to time and space.

It is straightforward to extend the Grzegorczyk Hierarchy using hierarchies of functions defined by recursion over tree-ordinals. We give the definitions and basic properties of the Fast-Growing Hierarchy and the Slow-Growing Hierarchy then state the relationship between them and the provably recursive functions of Peano Arithmetic. Further details may be found in [19], [43] or [51] .

Definition B.22. *For $\alpha \in \Omega^S$ the* ***Fast-Growing Hierarchies*** *of functions $F_\alpha : \mathbf{N} \to \mathbf{N}$ and $B_\alpha : \mathbf{N} \to \mathbf{N}$ are defined by the recursions*

$$\begin{aligned} F_0(n) &:= n+1, \\ F_{\alpha+1}(n) &:= F_\alpha^{n+1}(n), \\ F_\lambda(n) &:= F_{\lambda_n}(n), \end{aligned}$$

and

$$\begin{aligned} B_0(n) &:= n+1, \\ B_{\alpha+1}(n) &:= B_\alpha(B_\alpha(n)), \\ B_\lambda(n) &:= B_{\lambda_n}(n). \end{aligned}$$

Definition B.23. *For $\alpha \in \Omega^S$ the* ***Slow-Growing Hierarchy*** *of functions $G_\alpha : \mathbf{N} \to \mathbf{N}$ are defined by the recursion*

$$\begin{aligned} G_0(n) &= 0, \\ G_{\alpha+1}(n) &= G_\alpha(n) + 1, \\ G_\lambda(n) &= G_{\lambda_n}(n). \end{aligned}$$

Facts B.24. *(Fairtlough and Wainer [19]) For all $\alpha \in \Omega^S$*

1. G_α is increasing and strictly so when α is infinite. F_α and B_α are strictly increasing provided $n \neq 0$.

2. If $\beta \in \alpha[n]$ then $F_\beta(n) < F_\alpha(n)$, $B_\beta(n) < B_\alpha(n)$ and $G_\beta(n) < G_\alpha(n)$.

3. By parts 1 and 2, the hierarchies F_α, B_α and G_α form a majorization hierarchy on Ω^S. I.e., the functions are strictly increasing (except the constant functions $G_\alpha(n) = \alpha$ when α is finite) and each function at level α eventually dominates the corresponding function at level β for $\beta \prec \alpha$.

4. For $\alpha \neq 0$ and $n > 1$, $G_\alpha(n) < B_\alpha(n) < F_\alpha(n) < B_{\omega\cdot\alpha}(n)$.

For any total function f we let $E(f)$ denote the functions elementary in f. That is, $E(f)$ is the smallest class containing the initial functions, the addition function, the recursive difference function, the function f and is closed under formation of bounded sums and products.

Definition B.25. *(Löb and Wainer [37], [38] and Wainer [61]) For $3 \preceq \alpha \preceq \varepsilon_0$ the* ***Extended Grzegorczyk Hierarchy*** *is defined as*

1. If $\alpha := \beta + 1$ then $\mathcal{E}^\alpha := E(F_\beta)$.

2. If α is a limit ordinal then $\mathcal{E}^\alpha := \bigcup_{3 \leq \beta \prec \alpha} \mathcal{E}^\beta$

For finite α we see $\mathcal{E}^\alpha$ corresponds to the Grzegorczyk Hierarchy at and above the elementary functions. Furthermore $\mathcal{E}^\omega = PRIM$ and $\mathcal{E}^{\varepsilon_0}$ corresponds to the provably recursive functions of Peano Arithmetic. For the fast-growing B_α functions we find

1. $\bigcup_{\alpha \prec \omega \cdot i} E(B_\alpha) = \mathcal{E}^{i+2}$
2. $\bigcup_{\alpha \prec \omega^2} E(B_\alpha) = PRIM$
3. $\bigcup_{\alpha \prec \varepsilon_0} E(B_\alpha) = ProvRec(PA)$.

Proofs of these results may be found in [19].

The slow-growing functions G_α are *much slower* than F_α or B_α. Indeed $\bigcup_{\alpha \prec \varepsilon_0} E(G_\alpha) = \mathcal{E}^3$, the elementary functions. By the hierarchy comparison results of Girard [26] and later Cichon and Wainer [10], Wainer [62], we find that

1. $\bigcup_{\alpha \prec \phi(i,0)} E(G_\alpha) = \mathcal{E}^{i+2}$ for $i > 0$.
2. $\bigcup_{\alpha \prec \phi(\omega,0)} E(G_\alpha) = PRIM$.
3. $\bigcup_{\alpha \prec \varepsilon_0^*} E(G_\alpha) = ProvRec(PA)$.

where ϕ is the function defined by the Veblen Hierarchy and ε_0^* is a suitable tree-ordinal representation of the Bachmann-Howard ordinal (cf. [19]). Thus, following Arai [1], [2] and Schmerl [52], the Bachmann-Howard ordinal may be seen as the *slow-growing* or *pointwise* ordinal of Peano Arithmetic.

Bibliography

[1] Arai, T. *A slow growing analogue to Buchholz' proof*, Annals of Pure and Applied Logic. **54** (1991), 101–120.

[2] Arai, T. *Consistency proof via pointwise induction*, Archive for Mathematical Logic. **37** (1997), 149–165.

[3] Axt, P. *Iteration of primitive recursion*, Zeitschrift für Mathematische Logik und Grundlagen der Mathematik. **11** (1965), 253–255.

[4] Bellantoni, S. *Predicative recursion and the polytime hierarchy*, Feasible mathematics II (Ithaca, NY, 1992), Progr. Comput. Sci. Appl. Logic, vol. 13, Birkhäuser Boston, Boston, MA, 1995, pp. 15–29.

[5] Bellantoni, S. *Ranking arithmetic proofs by implicit ramification*, Proof complexity and feasible arithmetics (Rutgers, NJ, 1996), DIMACS Ser. Discrete Math. Theoret. Comput. Sci., vol. 39, American Mathematical Society, Providence, RI, 1998, pp. 37–57.

[6] Bellantoni, S. and Cook, S. *A new recursion-theoretic characterization of the polytime functions*, Computational Complexity. **2** (1992), 97–110.

[7] Bellantoni, S. and Niggl, K.-H. *Ranking primitive recursions: the low Grzegorczyk classes revisited*, SIAM Journal of Computing. **29** (1999), 401–415.

[8] Berger, U. *Program extraction from normalization proofs*, Typed Lambda Calculi and Applications (M. Bezem and J.F. Groote, eds.), LNCS, vol. 664, Springer-Verlag, 1993, pp. 91–106.

[9] Caporaso, S., Covino, E. and Pani, G. *A predicative approach to the classification problem*, Journal of Functional Programming. **11** (2001), 95–116.

[10] Cichon, E. A. and Wainer, S. S. *The slow-growing and the Grzegorczyk hierarchies*, Journal of Symbolic Logic. **48** (1983), 399–408.

[11] Cleave, J. P. *A hierarchy of primitive recursive functions*, Zeitschrift für Mathematische Logik und Grundlagen der Mathematik. **9** (1963), 331–346.

[12] Clote, P. *Ultrafilters on definable sets in arithmetic*, Logic colloquium '84 (Manchester, 1984), Studies in Logic and the Foundations of Mathematics, vol. 120, North-Holland, Amsterdam, 1986, pp. 37–58.

[13] Clote, P. *A safe recursion scheme for exponential time*, Proceedings of the 4th International Symposium on Logical Foundations of Computer Science (S. Adian and A. Nerode, eds.), Springer Lecture Notes in Computer Science, 1997, pp. 44–52.

[14] Clote, P. *Computation models and function algebras*, Handbook of computability theory, Studies in Logic and the Foundations of Mathematics, vol. 140, North-Holland, Amsterdam, 1999, pp. 589–681.

[15] Cobham, A. *The intrinsic computational difficulty of functions*, Logic, Methodology and Philos. Sci. (Proc. 1964 Internat. Congr.), North-Holland, Amsterdam, 1964, pp. 24–30.

[16] Covino, E. and Pani, G. *An implicit recursive language for the polynomial time-space complexity classes*, Journal of Universal Computer Science. **8** (2002), 75–84.

[17] Csillag, P. *Eine Bemerkung zur Auflösung der eingeschachtelten Rekursion*, Acta Univ. Szeged. Sect. Sci. Math. **11** (1947), 169–173.

[18] Çağman, N., Ostrin, G. E. and Wainer, S. S. *Proof theoretic complexity of low subrecursive classes*, Foundations of Secure Computation (Amsterdam) (F. L. Bauer and R. Steinbrueggen, eds.), NATO ASI Series F, vol. 175, IOS Press, 2000, pp. 249–286.

[19] Fairtlough, M. and Wainer, S. S. *Hierarchies of provably recursive functions*, Handbook of proof theory (S. R. Buss, ed.), Studies in Logic and the Foundations of Mathematics, vol. 137, Elsevier, Amsterdam, 1998, pp. 149–207.

[20] Feferman, S. *Predicativity*, The Oxford Handbook of the Philosophy of Mathematics and Logic (S. Shapiro, ed.), Oxford University Press, 2005, pp. 590–624.

[21] Feferman, S. and Hellman, G. *Predicative foundations of arithmetic*, Journal of Philosophical Logic. **24** (1995), 1–17.

[22] Ferreira, F. *A note on finiteness in the predicative foundations of arithmetic*, Journal of Philosophical Logic. **28** (1999), 165–174.

[23] Gaifman, H. and Dimitracopoulos, C. *Fragments of Peano's arithmetic and the MRDP theorem*, Logic and Algorithmic: An International Symposium held in honour of Ernst Specker, Monographie de L'Enseignment Mathématique, 1982, pp. 187–206.

[24] Gentzen, G. *Beweisbarkeit und Unbeweisbarkeit von Anfangsfällen der transfiniten Induktion in der reinen Zahlentheorie*, Mathematische Annalen. **119** (1943), 140–161.

[25] Gentzen, G. *The collected papers of Gerhard Gentzen*, North Holland, 1969.

[26] Girard, J. Y. Π^1_2*-logic part 1, Dilators*, Annals of Mathematical Logic. **21** (1981), 75–219.

[27] Goetze, B. and Nehrlich, W. *Loop programs and classes of primitive recursive functions*, Mathematical foundations of computer science, 1978 (Proc. Seventh Sympos., Zakopane, 1978), Lecture Notes in Comput. Sci., vol. 64, Springer, Berlin, 1978, pp. 232–238.

[28] Goetze, B. and Nehrlich, W. *The structure of loop programs and subrecursive hierarchies*, Zeitschrift für Mathematische Logik und Grundlagen der Mathematik. **26** (1980), 255–278.

[29] Grzegorczyk, A. *Some classes of recursive functions*, Rozprawy Mate. **4** (1953), 46.

[30] Hájek, P. and Pudlák, P. *Metamathematics of first-order arithmetic*, Perspectives in Mathematical Logic, Springer-Verlag, Berlin, 1993.

[31] Iwan, St. *On the untenability of Nelson's predicativism*, Erkenntnis. **53** (2000), 147–154.

[32] Kalmár, L. *Ein einfaches Beispiel für ein unentscheidbares arithmetisches Problem*, Mat. Fiz. Lapok. **50** (1943), 1–23.

[33] Kreisel, G. *On the interpretation of non-finitist proofs. II. Interpretation of number theory. Applications*, Journal of Symbolic Logic. **17** (1952), 43–58.

[34] Leivant, D. *Intrinsic theories and computational complexity*, Logic and computational complexity (Indianapolis, IN, 1994), Lecture Notes in Computer Science, vol. 960, Springer, Berlin, 1995, pp. 177–194.

[35] Leivant, D. *Ramified recurrence and computational complexity. I. Word recurrence and poly-time*, Feasible mathematics II (Ithaca, NY, 1992), Progr. Comput. Sci. Appl. Logic, vol. 13, Birkhäuser Boston, Boston, MA, 1995, pp. 320–343.

[36] Leivant, D. and Marion, J.-Y. *Ramified recurrence and computational complexity. II. Substitution and poly-space*, Computer science logic (Kazimierz, 1994), Lecture Notes in Comput. Sci., vol. 933, Springer, Berlin, 1995, pp. 486–500.

[37] Löb, M. H. and Wainer, S. S. *Hierarchies of number-theoretic functions. I*, Arkiv für Mathematische Logik und Grundlagenforschung. **13** (1970), 39–51.

[38] Löb, M. H. and Wainer, S. S. *Hierarchies of number-theoretic functions. II*, Arkiv für Mathematische Logik und Grundlagenforschung. **13** (1970), 97–113.

[39] Löb, M. H. and Wainer, S. S. *Correction: "Hierarchies of number-theoretic functions. I, II."*, Arkiv für Mathematische Logik und Grundlagenforschung. **14** (1971), 198–199.

[40] McCreight, E. M. and Meyer, A. R. *Classes of computable functions defined by bounds on computation: Preliminary report*, STOC '69: Proceedings of the first annual ACM symposium on Theory of computing (New York, NY), ACM, 1969, pp. 79–88.

[41] Meyer, A. R. *Depth of nesting and the Grzegorczyk hierarchy*, Notices of the American Mathematical Society. **12** (1965), 342.

[42] Nelson, E. *Predicative arithmetic*, Mathematical Notes, Princeton University Press, Princeton, NJ, 1986.

[43] Odifreddi, P. G. *Classical recursion theory. Vol. II*, Studies in Logic and the Foundations of Mathematics, vol. 143, North-Holland, Amsterdam, 1999.

[44] Oitavem, I. *New recursive characterizations of the elementary functions and the functions computable in polynomial space*, Revista Matemática de la Universidad Complutense de Madrid. **10** (1997), 109–125.

[45] Oitavem, I. *Implicit characterizations of Pspace*, Proof theory in computer science (Dagstuhl Castle, 2001), Lecture Notes in Comput. Sci., vol. 2183, Springer, Berlin, 2001, pp. 170–190.

[46] Ostrin, G. E. *Proof theories of low subrecursive classes*, Ph.D. thesis, Department of Pure Mathematics, University of Leeds, 1999.

[47] Ostrin, G. E. and Wainer, S. S. *Proof theoretic complexity*, Proof and system-reliability (Marktoberdorf, 2001) (J. Tiuryn and R. Steinbrüggen, eds.), NATO Sci. Ser. II Math. Phys. Chem., vol. 62, Kluwer Academic Publishers, Dordrecht, 2002, pp. 369–397.

[48] Ostrin, G. E. and Wainer, S. S. *Elementary arithmetic*, Annals of Pure and Applied Logic. **133** (2005), 275–292.

[49] Parikh, R. *Existence and feasibility in arithmetic*, Journal of Symbolic Logic. **36** (1971), 494–508.

[50] Parsons, C. *Ordinal recursion in partial systems of number theory (abstract)*, Notices of the American Mathematical Society. **13** (1966), 857–858.

[51] Rose, H. E. *Subrecursion: functions and hierarchies*, Oxford Logic Guides, vol. 9, The Clarendon Press Oxford University Press, Oxford and New York, 1984.

[52] Schmerl, U. *Number theory and the Bachmann/Howard ordinal*, Proceedings of the Herbrand symposium (Marseilles, 1981) (Amsterdam), Studies in Logic and the Foundations of Mathematics, vol. 107, North-Holland, 1982, pp. 287–298.

[53] Schütte, K. *Proof theory*, Springer-Verlag, Berlin, 1977, Translated from the revised German edition by J. N. Crossley.

[54] Schwichtenberg, H. *Rekursionszahlen und die Grzegorczyk-Hierarchie*, Arkiv für Mathematische Logik und Grundlagenforschung. **12** (1969), 85–97.

[55] Schwichtenberg, H. *Eine Klassifikation der ε_0-rekursiven Funktionen*, Zeitschrift für Mathematische Logik und Grundlagen der Mathematik. **17** (1971), 61–74.

[56] Schwichtenberg, H. *Content in proofs of list reversal*, Formal Logical Methods for System Security and Correctness (Amsterdam) (T. Nipkow O. Grumberg and C. Pfaller, eds.), Sub-series D: Information and Communication Security, vol. 14, IOS Press, 2008, pp. 267–285.

[57] Simmons, H. *The realm of primitive recursion*, Archive for Mathematical Logic. **27** (1988), 177–188.

[58] Tait, W. W. *Normal derivability in classical logic*, The Syntax and Semantics of Infinitary Languages (J. Barwise, ed.), Lecture notes in mathematics 72, Springer-Verlag, Berlin, 1968, pp. 204–236.

[59] Tait, W. W. *Finitism*, Journal of Philosophy. **78** (1981), 524–546.

[60] Visser, A. *Interpretability logic*, Mathematical logic, Proceedings of the Heyting 1998 Summer School in Varna, Bulgaria (P. Petkov, ed.), Plenum, New York, NY, 1990, pp. 175–209.

[61] Wainer, S. S. *A classification of the ordinal recursive functions*, Arkiv für Mathematische Logik und Grundlagenforschung. **13** (1970), 136–153.

[62] Wainer, S. S. *Slow growing versus fast growing*, Journal of Symbolic Logic. **54** (1989), 608–614.

[63] Wainer, S. S. and Williams, R. S. *Inductive definitions over a predicative arithmetic*, Annals of Pure and Applied Logic. **136** (2005), 175–188.

[64] Weiermann, A. *Sometimes slow growing is fast growing*, Annals of Pure and Applied Logic. **90** (1997), 91–99.

[65] Wilkie, A. J. and Paris, J. B. *On the scheme of induction for bounded arithmetic formulas*, Annals of Pure and Applied Logic. **35** (1987), 261–302.

[66] Williams, R. S. *Finitely iterated inductive definitions over a predicative arithmetic*, Ph.D. thesis, Department of Pure Mathematics, University of Leeds, 2004.

[67] Wirz, M. *Characterizing the Grzegorczyk hierarchy by safe recursion*, Master's thesis, Institut für Informatik und angewandte Mathematik, Universität Bern, 1999.

[68] Wirz, M. *Wellordering two sorts: a slow-growing proof theory for variable separation*, Ph.D. thesis, Institut für Informatik und angewandte Mathematik, Universität Bern, 2005.